

DFS-88

1988 TRI-SERVICE DATA FUSION SYMPOSIUM

TECHNICAL PROCEEDINGS

VOLUME I

JOHNS HOPKINS UNIVERSITY
APPLIED PHYSICS LABORATORY
LAUREL, MARYLAND

DISTRIBUTION STATEMENT A

Approved for public release
Distribution Unlimited

17 - 19 MAY 1988

SPONSORED BY :

JOINT DIRECTORS OF LABORATORIES
DATA FUSION SUB-PANEL
AND
CECOM CENTER FOR SIGNALS WARFARE
NAVAL AIR DEVELOPMENT CENTER
ROME AIR DEVELOPMENT CENTER
NAVAL OCEAN SYSTEMS CENTER

NOT FOUND
DOV CD

890001

W. Beamer
202B


*This sheet to
be printed
on inside cover*

Dear Reader,

On behalf of the distinguished members of the Joint Directors of Laboratories Data Fusion Subpanel (JDL/DFS), session chairmen and authors, it is indeed a pleasure for me to present to you the Technical Proceedings of the 1988 Tri-Service Data Fusion Symposium. The technical strength and uniqueness of this conference and these proceedings are directly attributable to the hard work and careful considerations of many talented people. The quality of the program sessions, technical paper content, and invited presentations is due entirely to their efforts. In structuring the conference we tried to create a program that was relevant, current, and functionally interrelated which encompassed the needs of the user, the researcher, the designer, and the developer. Accordingly, the papers covered the gamut from theory to application.

Our conference success, measured by the high quality of the papers, the caliber of the invited senior military speakers, the national and international prominence of our authors, presenters, and session chairmen combined with the overwhelming registration from all sectors of the technical community nationwide, has repaid us for all the time, effort, and concern exhibited during the conception and planning phases. I thank all of you sincerely for your early confidence, for your continued support, and for the very kind and laudatory comments we received. The JDL/DFS is already planning the DFS-89 agenda utilizing DFS-88 lessons learned plus fresh ideas, concepts, and imagination.

Thanks again for your continued interest and support. I hope this document provides for you a useful, lasting reference and memento of DFS-88.


Otto Kessler
Coordinator, DFS-88

TECHNICAL PROCEEDINGS
1988 TRI-SERVICE
DATA FUSION SYMPOSIUM

17 - 19 MAY 1988

SPONSORED BY:

JOINT DIRECTORS OF LABORATORIES
DATA FUSION SUBPANEL
AND
CECOM CENTER FOR SIGNALS WARFARE
NAVAL AIR DEVELOPMENT CENTER
ROME AIR DEVELOPMENT CENTER
NAVAL OCEAN SYSTEMS CENTER

19970605 055

PUBLISHED BY;

NAVAL AIR DEVELOPMENT CENTER
WARMINSTER, PA 18974

FOREWORD

The 1988 Tri-Service Data Fusion Symposium (DFS-88) was held at Laurel, Maryland on 17-19 May 1988 under the joint sponsorship of the Data Fusion Sub-Panel of the Joint Directors of Laboratories (JDL/DFSP), CECOM Center for Signals Warfare, Naval Air Development Center, Rome Air Development Center, and the Naval Ocean Systems Center. The symposium theme, THE ROLE OF DATA FUSION IN TACTICAL C², emphasizes (a) the emergence of technology for implementing data fusion in military systems, and (b) the growing recognition of the role of data fusion in the data collection/information generation processes required for situation awareness and threat assessment for both tactical and strategic purposes.

The purpose of the symposium was to provide a professional forum for the broad exchange of information and ideas on the development of data fusion technology for DOD applications. That we have made a significant first step in this direction is attested to by the quality of the technical papers and the level of interest evidenced by the large number of attendees drawn from a broad spectrum of disciplines. The symposium:

- o attracted 468 registered attendees (303 from industry and 165 from government [military/civilian]);
- o included operational C² briefs by Army, Navy, Air Force and Marine Corps representatives;
- o featured 36 technical papers dealing with research, development, and system applications of data fusion technologies and techniques; and
- o a commander's tactical C² perspective forum relating to multi-service experience involving command and control functions, information handling, and information fusion.

In publishing these Proceedings the sponsoring activities seek to facilitate an effective and timely dissemination of technical information. The Proceedings comprise two volumes. Volume I contains the unclassified papers available at the time of the symposium. Volume II contains the classified papers and unclassified papers received late. The papers contained in this document were printed directly from unedited reproducible copies submitted by the authors who are solely responsible for their contents.

THIS PAGE LEFT BLANK INTENTIONALLY

ACKNOWLEDGEMENT

The success of the symposium is a tribute to the speakers and authors and the parent organizations for their effort and interest in preparing these informative papers. Special acknowledgement is noted here to;

Mr. Frank White
Naval Ocean Systems Center

Mr. David Lange
National Security Agency

Mr. Richard Antony
CECOM Center for Signal Warfare

Dr. Jim Llinas
Calspan Corporation

Mr. Otto Kessler
Naval Air Development Center

Mr. Richard T. Baer
OADUSD The Pentagon

Mr. Len Converse
Rome Air Development Center

Dr. Clancy Hatleberg
Science Applications Int'l. Corp.

who collectively and individually made the symposium the success that it was. Their attention to details, both prior to and during the symposium, is responsible for the fine results.

THIS PAGE LEFT BLANK INTENTIONALLY

TABLE OF CONTENTS
VOLUME I
UNCLASSIFIED PAPERS

	<u>Page</u>
FOREWORD	i
ACKNOWLEDGEMENT	iii
TABLE OF CONTENTS	v
SYMPOSIUM ORGANIZATION	xiii
AGENDA	xv

KEYNOTE ADDRESS

RADM William O. Studeman, Director of Naval Intelligence Vol. II

TECHNICAL PAPERS - PRESENTED

Session A - OPERATIONAL C2 BRIEFS

BEKAA VALLEY CAMPAIGN 1982	1
Maj Gen Doyle E. Larson, USAF (Ret.)	

Session B - COMMANDERS TACTICAL C2 PERSPECTIVE FORUM

BG William E. Harmon - Army, Prog. Mgr. JTFPO	13
CDR Wayne Perras - Navy, FOSIC Officer CINCPACFLT	15

Session C - JOINT SERVICE/AGENCY PERSPECTIVES AND ISSUES

INTELLIGENCE SUPPORT TO DEPLOYED FORCES	37
CDR Scott Schneberger, Office of the Chief of Naval Operations	
DEFENSE SCIENCE BOARD REPORT ON COMMAND AND CONTROL SYSTEMS MANAGEMENT	55
Frank White, Naval Ocean Systems Center	

Session D - SPECIAL TOPICS I

DATA FUSION RESEARCH MANAGEMENT	75
Dr. Daniel F. Wiener II, University of Virginia	

	<u>Page</u>
THE COMPOSITE WARFARE COMMANDER'S TACTICAL DECISION SUPPORT SYSTEM	79
CAPT B. A. Thompson, SPAWARS	
LCDR C. T. Sutherlin, SPAWARS	
Dr. Arnold Hafner, SPAWARS	

Session E - KNOWLEDGE BASED TECHNIQUES

GETTING EXPERTISE INTO THE EXPERT FUSION SYSTEM	Vol. II
Edwin H. Miller, Sanders Associates	
BELIEF REPRESENTATION FOR FUSION USING EVIDENCE THEORY	91
John W. Betz, TASC	
PROACTIVE DATA FUSION FOR INTEGRATED TACTICAL WARNING AND ASSESSMENT	101
Joseph G. Wohl, ALPHATECH Incorporated	
MANAGING TEMPORAL UNCERTAINTY IN SITUATION ASSESSMENT	107
David Noble, Engineering Research Association	
SENSOR AND INFORMATION FUSION FROM KNOWLEDGE- BASED CONSTRAINTS	113
Allen R. Hansen, University of Massachusetts	
Edward M. Riseman, University of Massachusetts	
Thomas D. Williams, Amerinex Artificial Intelligence, Inc.	
DYNAMIC EVALUATION OF INFORMATION SOURCES IN RULE-BASED SENSOR FUSION SYSTEMS	123
H. James Antonisse and Karl S. Keller	
The MITRE Corporation	

Session F - METHODOLOGY IN DATA FUSION

INTEGRATION AND EXPERIMENTATION WITH COOPERATING EXPERT SYSTEMS, NATURAL LANGUAGE PROCESSING, AND ALGORITHMIC TECHNIQUES TO PERFORM CORRELATION AND DATA FUSION IN A TACTICAL ENVIRONMENT	131
James B. Nelson, Lockheed Missiles and Space Company	
DATA FUSION AS THE GUIDING PRINCIPLE IN THE BROAD AREA SEARCH PROBLEM	141
Dr. R. H. Cofer, Harris Corporation	
AUTOMATED SITUATION AND TARGET ANALYSIS	Vol. II
Joseph Antonik, Rome Air Development Center	

Session G - ADVANCED CONCEPTS

*AN ESM DATA FUSION PROCESS FOR TRACKING MASS RAIDS	149
Thomas M. Hart, the MITRE Corp.	
*AN APPLICATION OF COMPUTERIZED AXIAL TOMOGRAPHY (CAT) TECHNOLOGY TO MASS RAID TRACKING	155
John K. Barr, The MITRE Corp.	
A ROBUST SCHEME FOR FUSED TRACKING TBMs WITH NETTED SURVEILLANCE SENSORS	Vol. II
Clyde L. Humphrey and Debora M. Wesley Westinghouse Electric Corporation	
ARTIFICIAL NEURAL NETWORK ADAPTIVE SYSTEMS APPLIED TO MULTISENSOR ID	161
Dr. Christopher Bowman, VERAC	
A NEURAL NETWORK COMPUTATIONAL MAP APPROACH TO SENSORY FUSION	173
Dr. Clay D. Spence, John C. Pearson, Jack J. Gelfand, W. E. Sullivan and Richard M. Peterson David Sarnoff Research Center	
APPLICATION OF A CONDITIONAL EVENT ALGEBRA TO DATA FUSION	179
Dr. I. R. Goodman, Naval Ocean Systems Center	

Session H - TRACKER-CORRELATORS

A TECHNIQUE FOR AUTOMATICALLY CORRELATING ESM DATA AND RADAR TRACKS	189
George Rebovich, Jr. and Steven Bennett The MITRE Corporation	
MULTI-SENSOR TARGETING AND TRACKING	195
Dr. Charles P. Bernardin and Dr. Yun-Kung Jack Lin Texas Instruments	
ROBUST FUSION OF LOCAL INFORMATION	201
Raymond McKendall and Dr. Max Mintz University of Pennsylvania	

*These two papers were combined in the presentation entitled, "Mass Raid Tracking: CAT Technology and Data Fusion Processing"
Thomas M. Hart, John K. Barr, and George Rebovich, Jr.
The MITRE Corporation

	<u>Page</u>
A MULTI-FREQUENCY, MULTI-SENSOR DATA FUSION SYSTEM	209
John P. Lee and Dr. David Herring, ORINCON	
Dr. Joseph Fitchek, Naval Ocean System Center	
APPLIED PROBABILISTIC DATA ASSOCIATION	219
Jeff C. Brandstadt, GE Aerospace Electronic Systems	
MULTI-SPECTRAL/MULTI-SENSOR FUSION DEVELOPMENT FOR ENHANCING TARGET DETECTION AND TRACKING	Vol. II
Captain Martin E. Liggins, II, Rome Air Development Center	
Robert C. Washburn, ALPHATECH, Incorporated	
MULTIPLE SENSOR TRACKING OF CLUSTERS AND EXTENDED OBJECTS	231
Dr. O. E. Drummond, S. S. Blackman, and K. C. Hell	
Hughes Aircraft Company	

Session I - SPECIAL TOPICS II

COMMUNICATIONS IN COORDINATED TACTICS	239
Lee G. Morris, VEDA Incorporated	
ELECTRONIC WARFARE SITUATION ASSESSMENT PROCESSING REQUIREMENTS FOR FIGHTER AIRCRAFT OF THE 1990s	245
Ronald M. Yannone, General Electric Co. (INEWS JVT)	
IDENTIFYING ALIEN CONTACTS IN BAYESIAN CLASSIFIERS	251
D. G. Shankland, J. Johnson, D. W. Princehouse, and M. R. Smith	
Boeing Computer Services	
AN AUTOMATED TACTICAL SITUATION ASSESSMENT METHODOLOGY: ITS LIMITATIONS AND IMPACT UPON TACTICAL COMMAND AND CONTROL	257
Douglas Walter J. Chubb, CECOM Center for Signals Warfare	

Session J - PROGRAMS & APPLICATIONS

DATA FUSION IN ESAU: EXPERT SYSTEM AIR ORDER OF BATTLE UPDATE	269
Joyce Musselman, Greg D. Gibbons, Jill V. Josslyn, and John R. Russell	
Systems Control Technologies	
FROM LABORATORY TO THE FIELD: PRACTICAL CONSIDERATIONS IN LARGE SCALE DATA FUSION SYSTEM DEVELOPMENT	277
Gregory Fox, TRW Federal Systems Group	
Stephen Arkin, Space & Naval Warfare Systems Command	

	<u>Page</u>
REAL-TIME AUTOMATED TRACK MANAGEMENT	297
Susan J. Feldman, Hughes Aircraft Company	
DYNAMIC COORDINATE-TO-FEATURE ASSOCIATION FOR COMMAND AND CONTROL APPLICATIONS	307
Terence M. Cronin, CECOM Center for Signals Warfare	
TECHNICAL PAPERS - PUBLICATION ONLY	
DECISION AIDING: AN INTERACTIVE TRACKING AND IDENTIFICATION SYSTEM	315
Dr. Ivan Kadar and Dr. Manikant Lodaya Grumman Aircraft Systems	
MULTISENSOR ADAPTIVE CONTROL FOR THE ADI ENGAGEMENT PROCESS	319
Mac L. Hartless, GE Aerospace	
DATA REPRESENTATION AND MATCHING FOR EVENTS AND TEMPLATES	327
J. R. Gabriel, Argonne National Laboratory M. H. Gabriel, M. H. Gabriel and Associates	
A THEORY OF INTELLIGENCE ANALYSIS	333
J. R. Gabriel, Argonne National Laboratory M. H. Gabriel, Jet Propulsion Laboratory	
PREDICTIVE MODELS OF CORRELATOR/TRACKER ALGORITHM PERFORMANCE IN THE PRESENCE OF FALSE ALARMS	339
Barry Belkin, Gail A. Schweiter and Roberta S. Wenocur Daniel H. Wagner Associates	
DATA ASSOCIATION AND FUSION IN AN ASW ENVIRONMENT	347
Kenneth McPhillips and Robert Fagan Raytheon Company	
A MULTISENSOR APPROACH TO AIRCRAFT DETECTION AND IDENTIFICATION	357
G. A. Roberts and L. H. Bradford Ford Aerospace	
USE OF A TRACK-TO-TRACK ASSOCIATION ALGORITHM IN THE FUSION OF LONG-RANGE SENSOR DATA	369
David H. Kaskowitz, Robert G. Bryson and William H. Barker Tiburon Systems, Inc.	

	<u>Page</u>
A USER-SYSTEM INTERFACE DESIGN TOOL Stephen L. Morgan and Alexander Nauda HRB-Singer, Inc.	377
TRACK FILE FUSION Jeff Brandstadt, GE Aerospace	387
TRACK INITIALIZATION AS AN INTEGER PROGRAM WITH POSSIBLE TARGET DETECTIONS Scott L. Godfrey, FMC Corporation	397
STATISTICAL REASONING SYSTEMS Dr. Steven S. Noble, Boeing Advanced Systems	407
ALGORITHM FOR DECISION ON MERGING AND LINKING TARGET TRACKS J. B. Campbell, UNISYS J. E. Samaan, IBM	413
AUTOMATIC CORRELATION OF MULTIPLE INTELLIGENCE SOURCES Fred Schroeder and Larry Wright BTG, Incorporated	425
AN ADAPTIVE HULL-TO-EMITTER CORRELATOR Carey Priebe and David Marchette Naval Ocean Systems Center	433
THE KNOWLEDGE WORKBENCH: AN AI RAPID PROTOTYPE TESTBED Randall W. Hill, Jr., Jet Propulsion Laboratory	437
INTEGRATING PLANS AND SCRIPTS: AN EXPERT SYSTEM FOR INDICATIONS AND WARNING John W. Benoit, Edward J. Dombroski Pamela W. Jordan and Sharon J. Laskowski The MITRE Corporation	445
A TECHNIQUE FOR FUSING MULTIPLE SENSOR DATA. FOR TARGET MODELING AND EVALUATION Gary S. Mezynski and Karen S. Savage MIT Lincoln Laboratory	Vol. II
SMALL TARGET ENHANCEMENT WITH FUSED INFRARED (IR). AND RADAR DATA S. R. Yool, S. E. Stewart, J. R. Evans and D. N. Williams Naval Ocean Systems Center	Vol. II

ANGLE-ONLY MULTISENSOR CORRELATION AND TRACKING OF Vol. II
OBJECTS DURING MIDCOURSE BALLISTIC FLIGHT
R. B. Washburn, T. G. Allen, R. LaMaire and
H. Tsaknakis
ALPHATECH, Inc.

STATISTICAL-BASED MODELING OF SENSOR DATA COLLECTION Vol. II
D. L. Hall and R. J. Linn
HRB-Singer, Inc.

VOLUME II
CLASSIFIED PAPERS

Page

KEYNOTE ADDRESS

RADM William O. Studeman, Director of Naval Intelligence

TECHNICAL PAPERS - PRESENTED

MULTISPECTRAL MULTISENSOR FUSION DEVELOPMENT FOR
ENHANCED TARGET DETECTION AND TRACKING

Capt. Martin E. Liggins, II
Rome Air Development Center

A ROBUST SCHEME FOR FUSED TRACKING OF TBMS WITH
NETTED SURVEILLANCE SENSORS

Clyde L. Humphrey and Debora M. Wesley
Westinghouse Electric Corp.

GETTING EXPERTISE INTO THE EXPERT FUSION SYSTEM.
Edwin H. Miller, Sanders Assoc., Inc.

AUTOMATED SITUATION AND TARGET ANALYSIS.
Joseph Antonik, Rome Air Development Center

TECHNICAL PAPERS - PUBLICATION ONLY

A TECHNIQUE FOR FUSING MULTIPLE SENSOR DATA.
FOR TARGET MODELING AND EVALUATION

Gary S. Mezynski and Karen S. Savage
MIT Lincoln Laboratory

SMALL TARGET ENHANCEMENT WITH FUSED INFRARED (IR).
AND RADAR DATA

S. R. Yool, S. E. Stewart, J. R. Evans and D. N. Williams
Naval Ocean Systems Center

ANGLE-ONLY MULTISENSOR CORRELATION AND TRACKING OF
OBJECTS DURING MIDCOURSE BALLISTIC FLIGHT

R. B. Washburn, T. G. Allen, R. LaMaire and
H. Tsaknakis
ALPHATECH, Inc.

STATISTICAL-BASED MODELING OF SENSOR DATA COLLECTION

D. L. Hall and R. J. Linn
HRB-Singer, Inc.

APPENDIX

A. Registered DFS-88 Attendee List A-1

SYMPOSIUM ORGANIZATION

SPONSOR

Joint Directors of Laboratories Data Fusion Sub-Panel
CECOM Center for Signals Warfare
Naval Air Development Center
Rome Air Development Center
Naval Ocean Systems Center

HOST

John Hopkins University Applied Physics Laboratory

SYMPOSIUM CHAIRMAN

Dr. James H. Babcock, The Mitre Corporation

SYMPOSIUM CO-CHAIRMAN

Mr. Len Converse, Rome Air Development Center

SYMPOSIUM COORDINATOR

Mr. Otto Kessler, NADC

WELCOME ADDRESS

Dr. Carl Bostrom, Director Applied Physics Laboratory

SESSION CHAIRMEN

Col. John Rothrock, USAF
Mr. David Lange, NSA
Mr. Richard Baer, Intelligence Community Staff
Mr. Herbert Hovey, CECOM Center for Signals Warfare
Dr. Michael Drilling, ARI
Dr. David Hall, HRB-Singer
Dr. James Llinas, CALSPAN
Dr. Howard L. Wiener, CNO
Capt. Kurt Thomson, RADC
Mr. Edward Waltz, Allied/Bendix

TECHNICAL PAPERS SELECTION COMMITTEE

Mr. Frank White, NOSC
Mr. David Lange, NSAD
Mr. Richard T. Baer, OADUSD (I)
Mr. Len Converse, RADC
Mr. Joseph Mignogna, E-Tech

Mr. Otto Kessler, NADC
Mr. James Llinas, CALSPAN
Dr. Clancy Hatleberg, SAIC
Mr. Richard Antony, USACSW

THIS PAGE LEFT BLANK INTENTIONALLY

AGENDA

17 MAY 1988

Convene Symposium

Dr. James H. Babcock, MITRE Corporation, Chairman DFS-88

Introduction

Dr. James H. Babcock, MITRE Corporation, Chairman DFS-88

Welcome to Applied Physics Laboratory

Dr. Carl Bostrom, Director APL

Keynote Address

RADM William O. Studeman, Director of Naval Intelligence

Conference Overview

Dr. James H. Babcock, MITRE Corporation, Chairman DFS-88

Session A - OPERATIONAL C2 BRIEFS

Colonel John Rothrock, USAF, Session Chairman

BEKAA VALLEY CAMPAIGN 1982

May Gen Doyle E. Larson, USAF (Ret.)

PERSIAN GULF

CAPT Richard S. Loveland, Naval Operational, Intelligence Center

Session B - COMMANDERS TACTICAL C2 PERSPECTIVE FORUM

Moderator

Dr. James H. Babcock, MITRE Corporation, Chairman DFS-88

Participants

Col. Steven J. Whittenberger - USAF, Deputy Dir. for Oper. Support,
DCS Plans & Ops.

BG William E. Harmon - Army, Prog. Mgr. JTFPO

CDR Wayne Perras - Navy, FOSIC Officer CINCPACFLT

Brig. Gen Frank J. Breth - Hdqtrs. USMC

Session C - JOINT SERVICE/AGENCY PERSPECTIVES AND ISSUES

Mr. David Lange, National Security Agency, Session Chairman

Mr. Richard Baer, Intelligence Community Staff, Session Chairman

TIAP (Theatre Intelligence Architecture Plan)
LT Gen Edward J. Heinz, USAF; Dir. of Intelligence Community Staff

INTELLIGENCE SUPPORT TO DEPLOYED FORCES
CDR Scott Schneberger, Office of the Chief of Naval Operations

DEFENSE SCIENCE BOARD REPORT ON COMMAND/CONTROL MANAGEMENT
Frank White, Naval Ocean Systems Center

SIMNET (Simulation Network)
William Dejka, Naval Ocean Systems Center

ADJOURN

COCKTAIL RECEPTION
Kossiakoff Conference and Education Center, APL

18 MAY 1988

Convene Symposium
Dr. James H. Babcock, MITRE Corp., Chairman DFS-88

Introduction
Dr. James H. Babcock, MITRE Corp., Chairman DFS-88

Conference Framework
Frank White, Chairman Data Fusion Subpanel, JDL

Session D - SPECIAL TOPICS I

Mr. Herbert Hovey, CECOM Center for Signals Warfare, Session Chairman

DATA FUSION RESEARCH MANAGEMENT VIEWS FROM THE OTHER SIDE
Daniel F. Wiener II, University of Virginia

MULTI-LEVEL SECURITY
CAPT Thomas Herting, USN, National Computer Security Center

COMPOSITE WARFARE COMMANDER'S TACTICAL DECISION SUPPORT SYSTEM
CAPT B. A. Thompson, SPAWARS
LCDR LC. T. Sutherlin, SPAWARS
Arnold Hafner, SPAWARS

Session E - KNOWLEDGE BASED TECHNIQUES

Dr. Michael Drilling, Army Research Institute, Session Chairman

GETTING EXPERTISE INTO THE EXPERT FUSION SYSTEM
Edwin H. Miller, Sanders Associates

BELIEF REPRESENTATION FOR FUSION USING EVIDENCE THEORY
John W. Betz, TASC

PROACTIVE DATA FUSION FOR STRATEGIC C³ ATTACK WARNING
AND ASSESSMENT
Joseph G. Wohl, ALPHATECH Incorporated

MANAGING TEMPORAL UNCERTAINTY IN SITUATION ASSESSMENT
David Noble, Engineering Research Association

SENSOR AND INFORMATION FUSION FROM KNOWLEDGE BASED CONSTRAINTS
Allen R. Hansen, University of Massachusetts
Edward M. Riseman, University of Massachusetts
Thomas D. Williams, American Artificial Intelligence, Inc.

EXPERT FUSION SYSTEMS: REASONING, INTEGRATION, AND EVALUATION
H. James Antonisse, Karl S. Keller, John W. Benoit
Edward J. Dombroski, Pamela W. Jordon and
Sharon J. Laskowski
The MITRE Corporation

LUNCH

Luncheon Speakers
Frank White, NOSC, Chairman, Data Fusion Subpanel JDL
Dr. Clancy J. Hatleberg, Science Applications International Corp.

Topic: Data Fusion Community Issues and Future Directions

Session F - METHODOLOGY IN DATA FUSION

Dr. David Hall, HRB-Singer, Session Chairman

DATA FUSION FOR AUTOMATIC TARGET DETECTION AND LOCATION
Peter E. Cornwell and Daniel T. Bleck, Raytheon Company

INTEGRATION AND EXPERIMENTATION WITH COOPERATING
EXPERT SYSTEMS, NATURAL LANGUAGE PROCESSING,
AND ALGORITHMIC TECHNIQUES TO PERFORM CORRELATION
AND DATA FUSION IN A TACTICAL ENVIRONMENT
James B. Nelson, Lockheed Missiles and Space Company

DATA FUSION AS THE GUIDING PRINCIPLE IN THE BROAD AREA
SEARCH PROBLEM
Dr. R. H. Cofer, Harris Corporation

AUTOMATED SITUATION AND TARGET ANALYSIS
Joseph Antonik, Rome Air Development Center

COLLECTION REQUIREMENTS IN SUPPORT OF NAVY BATTLE MANAGEMENT
Joseph M. Moroney and Stanley F. Ralph, RCA ESD

Session G - ADVANCED CONCEPTS

Dr. James Llinas, CALSPAN, Session Chairman

MASS RAID TRACKING: CAT TECHNOLOGY AND DATA FUSION PROCESSING
Thomas M. Hart, John K. Barr, and George Rebovich, Jr.
The MITRE Corporation

A ROBUST SCHEME FOR FUSED TRACKING TBMs WITH NETTED
SURVEILLANCE SENSORS
Clyde L. Humphrey and Debora M. Wesley
Westinghouse Electric Corporation

ARTIFICIAL NEURAL NETWORK ADAPTIVE SYSTEMS APPLIED TO
MULTI-SENSOR ID
Dr. Christopher Bowman, VERAC

A NEURAL NET COMPUTATIONAL MAP APPROACH TO SENSORY FUSION
Dr. Clay Spence, David Sarnoff Research Center

APPLICATION OF A CONDITIONAL EVENT ALGEBRA TO DATA FUSION
Dr. I. R. Goodman, Naval Ocean Systems Center

ADJOURN

19 MAY 1988

Convene Symposium
Dr. James H. Babcock, MITRE Corporation, Chairman DFS-88

Introduction
Dr. James H. Babcock, MITRE Corporation, Chairman DFS-88

Conference Framework
Frank White, Chairman Data Fusion Subpanel, JDL

Session H - TRACKER-CORRELATORS

Dr. Howard L. Wiener, Office of Chief of Naval Operations, Session Chairman

A TECHNIQUE FOR AUTOMATICALLY CORRELATING ESM DATA WITH RADAR TRACKS

George Rebovich, Jr. and Steven Bennett
The MITRE Corporation

MULTISENSOR TARGETING AND TRACKING

Dr. Charles P. Bernardin and Dr. Yun-Kung Jack Lin
Texas Instruments

ROBUST MULTI-SENSOR FUSION

Dr. Max Mintz, University of Pennsylvania

TARGETING PICTURE ACCURACY MEASURE OF PERFORMANCE

David E. Mussmann
Johns Hopkins University/Applied Physics Laboratory

A MULTI-FREQUENCY, MULTI-SENSOR DATA FUSION SYSTEM

John Lee and David Herring, ORINCON
Joseph Fitchek, Naval Ocean Systems Center

APPLIED PROBABILISTIC DATA ASSOCIATION

Jeff C. Brandstadt, GE Aerospace Electronic Systems

MULTISPECTRAL/MULTISENSOR FUSION DEVELOPMENT FOR ENHANCING TARGET DETECTION AND TRACKING

Captain Martin E. Liggins, II, Rome Air Development Center
Robert C. Washburn, ALPHATECH, Incorporated

MULTIPLE SENSOR TRACKING OF CLUSTERS AND EXTENDED OBJECTS

Dr. O. E. Drummond, S. S. Blackman, and K. D. Hell
Hughes Aircraft Company

TRACKER/CORRELATOR PANEL DISCUSSION

Session I - SPECIAL TOPICS II

Captain Kurt Thomson, Rome Air Development Center, Session Chairman

COMMUNICATIONS IN COORDINATED TACTICS

Lee G. Morris, VEDA Incorporated

ELECTRONIC WARFARE SITUATION ASSESSMENT PROCESSING REQUIREMENTS FOR FIGHTER AIRCRAFT OF THE 1990s

Ronald M. Yannone
General Electric Company (INEWS JVT)

IDENTIFYING ALIEN CONTACTS IN BAYESIAN CLASSIFIERS

D. G. Shankland, J. Johnson, D. W. Princehouse, and M. R. Smith
Boeing Computer Services

THE DESIGN OF A COMBAT INFORMATION PROCESSOR

Dr. Philip J. Emmerman, Harry Diamond Laboratories

AN AUTOMATED TACTICAL SITUATION ASSESSMENT METHODOLOGY:

ITS LIMITATIONS AND IMPACT UPON TACTICAL COMMAND AND CONTROL
Douglas Walter J. Chubb, CECOM Center for Signals Warfare

Session J - PROGRAMS & APPLICATIONS

Edward Waltz, Allied/Bendix, Session Chairman

DATA FUSION IN ESAU - AN EXPERT SYSTEM FOR AIR ORDER
OF BATTLE ANALYSIS UPDATE

Joyce Musselman, Greg D. Gibbons, Jill V. Josselyn,
Janice A. Malin, John R. Russell, and Peter Bernstein
Systems Control Technologies

FROM LABORATORY TO FIELD: PRACTICAL CONSIDERATIONS
IN LARGE SCALE DATA FUSION SYSTEM DEVELOPMENT

Gregory Fox, TRW Federal Systems Group
Stephen Arkin, Space & Naval Warfare Systems Command

REAL-TIME AUTOMATED TRACK MANAGEMENT

Susan J. Feldman, Hughes Aircraft Company

DYNAMIC COORDINATE-TO-FEATURE ASSOCIATION FOR COMMAND
AND CONTROL APPLICATIONS

Terence M. Cronin, CECOM Center for Signals Warfare

SUMMARY AND WRAP-UP

Dr. James H. Babcock, Chairman DFS-88

TECHNICAL PAPERS

THIS PAGE LEFT BLANK INTENTIONALLY

BEKAA VALLEY CAMPAIGN 1982

Maj Gen Doyle E. Larson, USAF (Ret)

I want to talk to you today about the Bekaa Valley Campaign of 1982 from the viewpoint of the Combatant Element Commander. The perspective is from the Corps division level and other levels in my perception of what happened in the Bekaa Valley. At the time this occurred, almost six years ago, I was not only the Commander of the Electronic Security Command, but I was the Director of the Joint Electronic Warfare Center. It was my responsibility to keep the Joint Chiefs informed on all things "electronic"--that is, electronic warfare, electronic combat, C³ countermeasures--etc. After the incursion into Lebanon I began working very hard with the National Security Agency to examine and assess the data that we had, and with DIA to find out exactly what had happened in the Bekaa Valley. Even after this data analysis, there were still operational issues that were unanswered. What did the Commander attempt to do? What kind of tactics did he use? And so I sent one of my officers over to Tel Aviv to try to get answers to this question of tactics and operations, and to my dismay when he returned he had been told either lies or nothing, which is sort of good Israeli OP SEC. As a result, when I retired in 1983, I had an excellent idea from an intelligence standpoint what had happened in the Bekaa Valley, but the operator view was still missing. Fortunately I was invited over to Israel, spent ten days with the Israeli Defense Forces up into the Bekaa Valley for a day and a half, and down into the Sinai where I had the good fortune to run into the Chief of Israeli Intelligence Production during the incursion, and had a chance to talk about the situation. When it kicked off on 9 June 1982 it had culminated several years of planning. I became aware of that because of the fact that the OP's order, Peace For Galilee, was signed by Ezer Weitzman. He hadn't been the Minister of Defense for over two years when the incursion took place which told me immediately that the Israeli Air Force had that plan, Peace For Galilee, in their hands for over two years and I'm sure had been practicing and drilling for the execution. And so when the day finally came when the Israeli government had had

all they could take from the PLO and they determined they were going to do something about it, they issued the order to Maj Gen David Evri, the Commander of the Israeli Air Force, to begin execution of Peace For Galilee. So he drove up the road to his command post at Mt. Maron and began executing a plan that he had been working on for a long time.

Now going back at least two years then, the first thing David Evri recognized was that to drive the PLO out, destroy their base, and defeat the Syrian forces, you are going to have to move some columns of Armor infantry North, and he was going to have to provide air support for that operation. And as he thought about that, David Evri said, "I am not going to make the mistake that we made in Yom Kippur--before we go after tanks and artillery and troops, we are going to suppress the air defense and we are going to gain air superiority." You recall in Yom Kippur in the first five days they lost one-third of their Air Force. And so, David Evri said, we're not going to do that in Peace For Galilee. We are going to go in there and get control of that air space first and foremost and then we'll support tanks. And so he met with his staff and he informed his staff that he wanted to exercise what he called command and control warfare. He was going to attack the Syrian's ability to control their forces. That was his game plan and he wanted to use a disruptive strategy, which to David Evri meant he wanted to be able to combine deception and jamming with his lethal capability. He identified, essentially, four tools that he intended to use in that incursion. He wanted to use his lethal capability, of course, but he also wanted to jam and use some deception in a number of ways. However, the critical part of this strategy, the tool that is most important, and which must be in place at the outset, is what he called intelligence exploitation. But he said I want to exploit the enemy's command and control systems and identify the weaknesses that I can exploit. And so he met with his intelligence people and they began to show General Evri what was in the database. They told him that they had a pretty good handle on artillery, self-propelled

artillery in particular, and major elements which have been located. Their database on infantry was pretty complete--down to fairly small sized units--although precise locations were not available. They also had a pretty good handle on armor but did not know where the division bunker was, nor their headquarters. And then, of course, Evri had mentioned air defense as a number one consideration. And they were very proud to tell General Evri that each week they surveyed each prepared site and each SAM site with imagery so that they had each site down to one meter in accuracy and they felt very good about their locational information and so they ended up very proudly showing him what was in the Bekaa Valley.

General Evri said, "Well, that's very nice guys, but that isn't going to support my strategy." Furthermore he said, "I don't have enough bombs and bullets to take care of all the combatant elements and I certainly can't jam everything. He said, "I want you to go back to the drawing board, and I want you to do a nodal analysis. I want nodal intelligence." They came back eventually and they had the nodes. Between each one of the nodes for armor they identified the type of communication link hooking those nodes together. They showed this to General Evri and as a matter of great pride they said in the process of doing this nodal analysis they ended up putting imagery and SIGINT together (fusing it) and found the bunker, and it was only when we did that that they could find that bunker. General Evri was then interested in the red line going down to the tank maneuver elements and he said "What is that red line?" They said that's a Soviet built low VH radio. He said, "You mean to tell me that's the only way they communicate at that level?" And they said, "Yes, it is." And they said, here's the surface-to-air missile and Triple A problem and again the nodes were explained. Once again the single red line caught David Evri's eye and he said, "You mean to tell me that the launcher receives tracking data from higher echelons and receives its command and control only on that low VH line?" And they said, "Yes." And then he said, "I assume that's an encrypted line?" And

they said, "No, it's unencrypted." He said, "Well that's very interesting." "What's the frequency range?" They said between 30 and 50 MHz. Now that was the same Bekaa Valley when they got through this nodal analysis. Same number of combatant elements, only now David Evri was only looking at a hand full of nodes and a single red line.

Then David Evri said something else to his intelligence people. He said, "Every sensor that you manage has to do two jobs simultaneously." On the right hand side of the chart Evri said I want you to correlate and fuse, validate and analyze, and give me the best all-source data you can each and every day up until the battle starts and I am going to use this for my major planning and for targeting of fixed facilities. And certainly at the end of each day's battle, I want to have battle damage assessment so that I know what I need to adjust based upon every source you've got available before I start tomorrow's battle." But he said, "When the balloon goes up, each sensor you're operating has also got to deliver the left side of the chart for me also, that is, essentially locational data. I want you to tell me what is it, where is it, and when did you detect it, and I want it as fast as you can give it to me. And you can tell me how inaccurate it is if you want to but I have to have it fast. If you can't meet the speed requirement, give it to me at the end of the day in the all-source, fused, analyzed form. But in order to use information for real-time targeting and to chase mobile things, things that scoot and shoot on the field of the Bekaa Valley, I want in some cases single sensor data as fast as you can get it to me. And I will live with the inaccuracies if you will just tell me what they are. Let me make judgments on how I can use the data.

The day came when Evri was told to execute Peace For Galilee and so he began launching his intelligence assets. Some of my slides now were not taken in Israel. Some of them show U.S. comparable equipment. The U.S. did in fact build, E-Systems, Greenville, built an excellent ELINT aircraft with a data link right into the Mt. Maron command post.

It was an important asset. Evri put it in place early on that day. ESL has built them a GUARDRAIL system; it does an excellent job. The receivers are controlled on the ground. An again the data was on the ground and was available to Evri directly. Probably the most important intelligence support for General Evri, however, was on top of Mt. Hermon where since the 1967 war the Israelis have occupied the high ground and have some excellent COMINT and ELINT equipment up there. Looking to the east you can see all the Syrian airfields. Line of sight coverage is very good and looking to the west you can see all the way down to downtown Beirut. So he had excellent line of sight coverage. The Mt. Hermon command bunker is located just below the "e" in Israel on the slide. When you drive past it, it is not a very prominent peak, but it is high enough so that Evri had connectivity with his intelligence sensors and he had connectivity with his jammers and with the rest of his command and control assets. I asked permission each day of the ten days I was there to visit Mt. Hermon and each morning they would sound very promising that they would take me there, but I never got there. So I have to imagine what it looks like inside. I know that they were receiving a lot of voice reports. I know they were receiving some digital data and display, but I suspect they had a combination of very fast grease pencil displays and some digital displays and the voice mostly coming down from Mt. Hermon to Mt. Maron provided the basis for what was being plotted here. When he had his intelligence sensors up and looking at those things that they knew he wanted to look at first, he began launching his aircraft.

Here is a case where he decided to use some deception. He was obviously going to launch every aircraft silently-- "IFF, no radar, no radio communications, launch with flares only." But just to make sure that the U.S. RC135 and the P-3s operating out over the Med didn't detect a rather large size operation underway, he also launched a bunch of training missions using operational IFF codes and call signs. And to our linguists onboard the RC135 that morning, the morning seemed

fairly routine. The 247 aircraft got in place and began low-level orbiting and with the time set up just right, which had been practiced in the Sinai, at the precise minute Evri launched some special forces to go after the SAM sites in the Bekaa Valley. When the timing was even a little more close to the actual kick off of the combat, he then began launching some artillery rounds at the southern most Syrian units. So he was getting pretty good suppression of air defense from his artillery. And then finally Evri began launching some minidrones. This Melpar drone looks an awful lot like the Teddirane drone that I watched being manufactured in the factory in Tel Aviv. Evri had a couple of purposes for these drones. I would say he had three reasons to use drones. The first was, he had some excellent ELINT equipment, but you can't collect ELINT if the enemy doesn't emit. And so he wanted to provoke the enemy into using his radar. So one of the reasons for having drones was to get those emitters up on the air so he could confirm their location and do something with it. He used the drones also for deception. They put radar repeaters in the drones which made them look just like K-4's and F-4s. And then he had a third purpose for the drones. And that was to carry video electro-optical and capability which was controllable from his command post. This shows you the actual control position in the Mt. Maron bunker with a map on the right hand side showing the current location of the drone and on the left hand side the controls and what is being seen by the camera. This permitted Evri to see in real-time and match or fuse imagery with his SIGINT coming into him from multiple sources. Excellent, all-source fused information.

It was about that time that the linguist on top of Mt. Hermon began responding to David Evri's specific guidance. He said I want you to look at the SAMs first and when you hear them controlling their launchers, I want you to repeat verbatim what you hear. I don't want you to analyze it or summarize it. I want you to give it to me as you get it by voice reports. And so what the linguist began to hear on those unencrypted links were the Syrian's taking the deception bait.

They saw the drones coming in from the south and they began chattering at their launch operators that in fact the Israelis were beginning their attack. The K-4s and F-4s were coming in from the south and they issued firing instructions. And as soon as Evri heard those instructions being given he knew they had taken the bait so he wanted to make damn sure they didn't get a countermanded order and so he turned on the jammers. Between 20 and 50 MHz onboard the RC135, the panoramic display scope looked milky white with noise. No launch operator was going to get anymore information from up channel nor was he going to get any other tracking data. He was pretty much on his own. He was going to have to use his own radar and his own optical tracker.

At that point Evri issued one code word--ATTACK to his 237 aircraft and at 3 o'clock in the afternoon at 4,000 to 6,000 ft. the attack started. If you look toward the west from Bekaa Valley at 3 o'clock in the afternoon I guarantee you will see a very bright sun and your optical tracker will be absolutely useless and that's exactly what happened. In ten minutes they destroyed every SAM site in the Bekaa Valley. I talked to a lot of the pilots who were involved in the operation. I looked at a lot of the film. It looked a lot to me like they used dumb bombs. I saw neat holes in every control van, every launcher. I could see no missiles on any rails. In fact the pilots reported not seeing any missiles. It appears that they had pickled off everything they had against the drones and were reloading when they caught them. As soon as Evri issued that code word for the attack to start, Col Rahm, his Intelligence Production Chief, knew immediately that Evri was no longer interested in SAMs. He had done everything required to destroy those SAM sites. David Evri's information displays were changed immediately because he was not focusing on air threat, Flogger and Migs. And that code word to start the attack was a signal to his linguist on top of Mt. Hermon to start monitoring fighters as a first priority. And again the explicit instructions that he had given those linguists was the same--report exactly what you hear as you hear it. I want it verbatim. What they began to

hear on encrypted VH communication links were the controllers talking to the lead pilots about the attack going on in the Bekaa Valley and issuing specific guidance to those Migs and Floggers. What happened was that the senior GCI controller made a very grave tactical mistake. Out over the Med at about 5,000 to 6,000 ft. he saw another wave of aircraft and he concluded that that was the second wave of ground attack aircraft and so he advised his Floggers and Migs to execute the blow-through tactic. This is the same tactic the Russians taught the Vietnamese. Come in on the deck, pull up sharply through the attacking force and disrupt their bombing attack.

As soon as Evri heard the blow through command being given, the jammer started. And again onboard the RC135 on a panoramic display between 130 and 150 MHz it was very white. If you were flying wing on a Flogger, you could probably talk to your wingman. But you weren't going to talk to anybody on the ground. You weren't going to talk to anybody more than a few miles away. So essentially the last instruction had been given to those Floggers and Migs and sure enough into the Bekaa Valley they went on the deck, pulled up sharply in front of the combat air patrol. F-16s just waiting for them. In ten minutes they shot down 29 aircraft. Down in the Sinai at the new Ramone Air Base I took a look at a lot of the F-16 film-gunner film. Every film clip looked about the same. Three to five seconds in length. The first thing you would see on the F-16 heads up display was a Flogger or a Mig at the outside edge. You would see the wings go up pretty sharply and you would see an AIM-9L come off the rail and splash on each Mig or Flogger. In fourteen minutes of elapsed time, they had destroyed 19 SAM sites and shot down 29 aircraft.

For all intents and purposes, you could say David Evri had air superiority. He went on to shoot down a total of 80 aircraft, lost one F-4 when the pilot got a little too slow at low altitude chasing a ground target and an SA-7 got him. Evri manipulated his information. He manipulated those tools extremely well. Now the Joint Chiefs looked

at that and declared that it was a classic case of C³ countermeasures. Evri glued these four tools together with OP SEC. He was smart enough to recognize if you don't have OP SEC, you don't have deception except against your self. He's combat experienced enough to know that you don't have effective jamming if your SIGINT system isn't tethered right to your jammers because you are going to be jamming the wrong things. You are not going to be effectively jamming and so you are just going to be sending electrons out in the air. And finally, exploitation. Evri controlled the sensors for that battle. He did not go to Tel Aviv and appear before a committee to plead for intelligence. When he said I want you to listen to SAMs now, they listened to SAMs. When he said move onto airplanes, they moved onto airplanes. When he said I want you to report verbatim, they reported verbatim. Obviously as you go up in complexity in command arrangements it gets to be more difficult.

The sad state of the truth in the U.S. is that today for the air-land team which I am trying to depict here--the Army, Air Force, air-land stream or for that matter Navy, Marine power projection, we provide damn little information like Evri got on the left hand side of that chart. In fact, I suggest to you that only the air defense mission for the Air Force gets that kind of support. Air-to-mud missions get zip. The fighter wings and the divisions get close to zip. Most of the stuff--I was just in Korea for Team Spirit and the three star up at Camp Red Cloud was about three feet off the ground upset with his intelligence support. He said they come rushing in at the end of the battle with a summary, an intelligence summary; this report tells me how I lost, he says---during the day I got nothing. And that is, generally speaking, the state of affairs. Here's kind of the situation: Our intelligence people are working diligently but maybe we have too many managers. Maybe DIA has the charter but not the capability. You can't transfer our intell databases easily, and you can't access them easily. And there certainly aren't any nodal analyses. There is an effort underway in San Antonio called COMFY

WEB which started out as a genuine effort to provide a C³ database and nodes. It is now so unwieldy, so large that you would have to carry a huge computer to the field with you to do anything with it. Furthermore, the SIGINT system can't feed it because there are too many data elements in there. In point of fact, I think it is absolutely excessive to have more than the 19 data elements that NATO has agreed to exchange in that database. Speaking for the commander conducting command and control warfare, he only needs to know what is it, where is it, and when did you see it. And he can find out frequency information pretty well on his own. Now he also needs to get some stuff fast. When I say fast obviously intelligence sensors can't report to the commander until they've got resolved data, i.e., they can't get it until they capture it. But once they do capture it instead of typing up a paper report, it would be very nice if they would just click out a symbol indicating the location of that threat or target and then type up the paperwork. What happens today is we have "Cadillac" collectors. I mean the finest collection systems the world has ever seen operating in a digital world. But when it hits the ground the information largely goes into record communications. It becomes a piece of paper and goes out of the digital stream. It takes time to poke up messages. It takes time to transmit messages. Generally speaking in Team Spirit the air commanders were getting their information in 27 minutes. Up to Red Cloud it was two hours. It's tough to conduct armed warfare with two hour old information. When it gets to the bunker it ends up on the end of a grease pencil for most bunkers.

Now there are a couple of things I think that we have to do in this information business, fusion business, intelligence support business. The first thing is we've got to recognize who is it we are supporting. There is a commander, there is a decision maker for every mission area and he is the one who needs to give us explicit guidance instructions on what he needs, what he wants. And for each mission on the battlefield the needs are a little bit different. Secondly, the commander should transmit that data to his sensor operators in

graphic form. In the U.S. Air Force today we issue an air tasking order. It generally is about 26 pages of teletype. The Intelligence Manager generally needs about one-half page of data out of that 26 pages. By the time you grind 26 pages through the Communication Centers I can guarantee you that the operations have started before the intelligence guys know about it. And the only thing that Intelligence Sensor Manager needs really is a graphic picture. This is the view of the Corps Commander's mind of the battle that is going to unfold. And I get a lot of flack from my Air Force when I say that the Corps Commander is the quarterback for the air-land team, but by God he is! You are not going to conduct air-to-mud missions in his area unless he tells you to, unless he asks you to. And so his vision of that battle needs to be transmitted to that Air Commander. And they need to discuss all the areas where air support is essential to the Corps Commander. He needs intelligence support. He can't see deep enough. He needs jamming. He doesn't have assets that are high enough. He doesn't have enough jammers. He needs deception support. He doesn't have enough inventory. And obviously he needs close air support and interdiction support. And they need to discuss it throughout the battle area, close-up, deep throughout the battle day. And this needs to be translated into action. But it should start out with a graphic. And when that fighter wing has received that graphic and has been tasked to do an interdiction job, and he finishes his penetration planning, he should send the Intelligence Sensor his graphic. This is where I am going, this is the time I'm going, this is what my database tells me is there. And you sensor operators, if you've got changes, I need to know about it whether they are targets or threats. Now here is a device that I think is going to be very useful. It is called Senior Jade. Air Force Logistics Command is building it now. They have a prototype at Metro Tango. It's been there for several years. This is an automated correlation system that automatically correlates different digital streams. It will start out correlating radar and COMINT or ELINT. But in point of fact, it could be used to correlate any type of digital stream and tell you very quickly from as many different sources as possible

what is it, where is it, and when did you detect it and give you some identity information. The second thing we've got to do is get JTIDS. It seems to me that we will never be able to conduct combined arms warfare in a NATO environment until we get an ability to operate a disseminated command and control system on the battlefield that is secure and jam resistant. And if this doesn't look like the best approach, I'll eat my hat for sure. I think it is the wave of the future. I'm delighted to hear words at the National Security Agency that they are planning on it for their architecture. And finally, DARPA and NSA developed a little device here that is terribly useful to a tactical commander. It is called Fulcrum software. It essentially uses analog stored DMA maps and superimposes sensor data on top of that. It will change maps in a third of a second. The registration of the sensor data is more accurate than the sensors themselves. It is off the shelf. We have a little over 350 systems out in use today, and it is cheap. I think the TEMPEST version is less than \$40K total.

Now a lot of people say to me, well General Larson that resolution is not very good. Well my reply is, "Generals don't have very good eyes." Furthermore, "Generals don't need to be looking at so much detail." Let the weapons controllers take care of that detail. This will present in quite good resolution on real maps the symbols that the general is interested in keeping track of. So I think those are three extremely important developments coming along. And with that point I'll quit and let you hear something about the Persian Gulf.

BG William E. Harmon, Army
Program Manager, JTFPO

I was asked to discuss data fusion from a user perspective. My military background is principally that of an intelligence officer in the field of tactical intelligence. As the current Program Manager of the Joint Tactical Fusion Program, I have responsibility for a wide range of Army and Air Force fusion programs, with the major program consisting of the development of the Army All-Source Analysis System (ASAS) and the Air Force Enemy Situation Correlation Element (ENSCE). These systems will provide automated intelligence support to tactical Army and Air Force commanders. They will support the operations and fire support elements as well as the intelligence officer and will provide an integrated picture of the battlefield.

ASAS/ENSCE is an evolutionary development program because the problem it addresses is too big to be solved in one increment. ASAS/ENSCE has evolved and benefits from a number of other programs I also manage. Most of these programs are already in the field; the resulting dynamic user feedback makes my job more challenging but is essential if an evolutionary program is to deliver useful systems.

The LOCE (Limited Operational Capability Europe) and LENSCE (Limited ENSCE) systems are major contributors to the ASAS/ENSCE program. These systems, derived from the BETA (Battlefield Exploitation and Target Acquisition) development, have evolved over years of extensive field use. At Gallant Knight 86, using LENSCE, for the first in my military career I saw three and four-star Generals really in sync on exactly what they wanted to do and how they wanted to fight the battle. The LOCE correlation algorithms are being used in the baseline ASAS/ENSCE.

I am responsible for the Tactical Simulation (TACSIM) System which generates thousands of messages for theater-level exercises and serves as the message driver for ASAS/ENSCE testing. The Technical Control and Analysis Center (TCAC) systems provide automated communications handling, message processing and data retrieval in support of tactical SIGINT operations. Twenty shelters are deployed in divisions and corps in Europe and Ft Bragg, with six more being built for the Marine Corps. Like LOCE/LENSCE, TCAC's success is due in large part to user feedback. I have oversight of the Army's Microfix program which uses Apple Computers.

We are involved in the fusion architecture at theater level to insure that we field systems that are compatible with theater systems and requirements. Under the Fusion Analysis Technology program, we monitor advanced technology efforts in your laboratories, NSA, CIA and DARPA, and we evaluate potential applications for planned product improvements to ASAS/ENSCE. The

Digital Topographic Support System (DTSS), run by the Engineering Topographic Laboratory, will provide line-of-sight and terrain analysis map products required by ASAS/ENSCE. We recently assumed responsibility for development of the Integrated Meteorological System (IMETS).

The field commander needs to know enemy, weather, and terrain. The objective of ASAS/ENSCE is to dynamically integrate this information and present it to him in concise terms of the enemy situation and intentions so that he can make decisions to effectively employ combat power. Most importantly, he should understand what he does not know about the battlefield in time to intelligently direct his collection assets.

Today we have efficient sensors but not very efficient processing of sensor collections. Current largely manual systems get bogged down in message handling. ASAS/ENSCE will provide quick communications processing and dynamic data base storage, manipulation and display. It won't replace the analyst, but it will provide analytical aids to facilitate decision-making.

An evolutionary program requires a sophisticated audit of requirements on a continuing basis. The physical ASAS/ENSCE has been reduced from 10-ton trailers to S-250 shelters with smart, portable workstations. The baseline hardware for ASAS/ENSCE has been field tested. The first three software baseline releases have been established from user conferences and prioritization of functionality based on cost/schedule risk and benefit analyses. The first software release will be field tested in March of next year. Requirements not incorporated in the baseline releases will be reprioritized, taking into account emerging technology, for future upgrades. This is our roadmap to achieve the ASAS objective - to allow the commander to fight an effective air-land battle.

COMMANDER'S TACTICAL C2 PERSPECTIVE

BY

**CDR WAYNE PERRAS, USN
CINCPACFLT
HEAD INTELLIGENCE PLANS, PROGRAMS & POLICY**

OVERVIEW

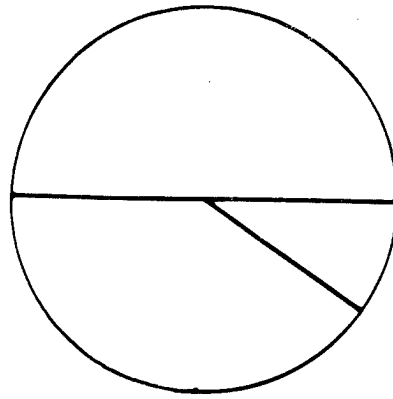
- FUSION PERSPECTIVE
- PACOM PERSPECTIVE
- PACFLT PERSPECTIVE
 - FUSION ASHORE
 - FUSION AFLOAT
- WARFIGHTING REQUIREMENTS SCENARIO

FUSION

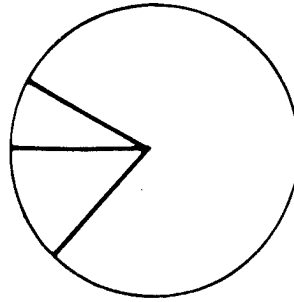
- MERGING
- DIFFERENT ELEMENTS ...
- UNIFIED WHOLE

COMPARISONS

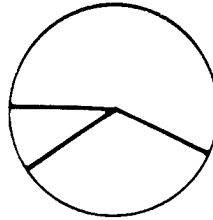
■ SIGINT
 ■ HUMINT
 ■ IMINT



SOUTHEAST ASIA



PHILIPPINES



INDIA/SRI LANKA

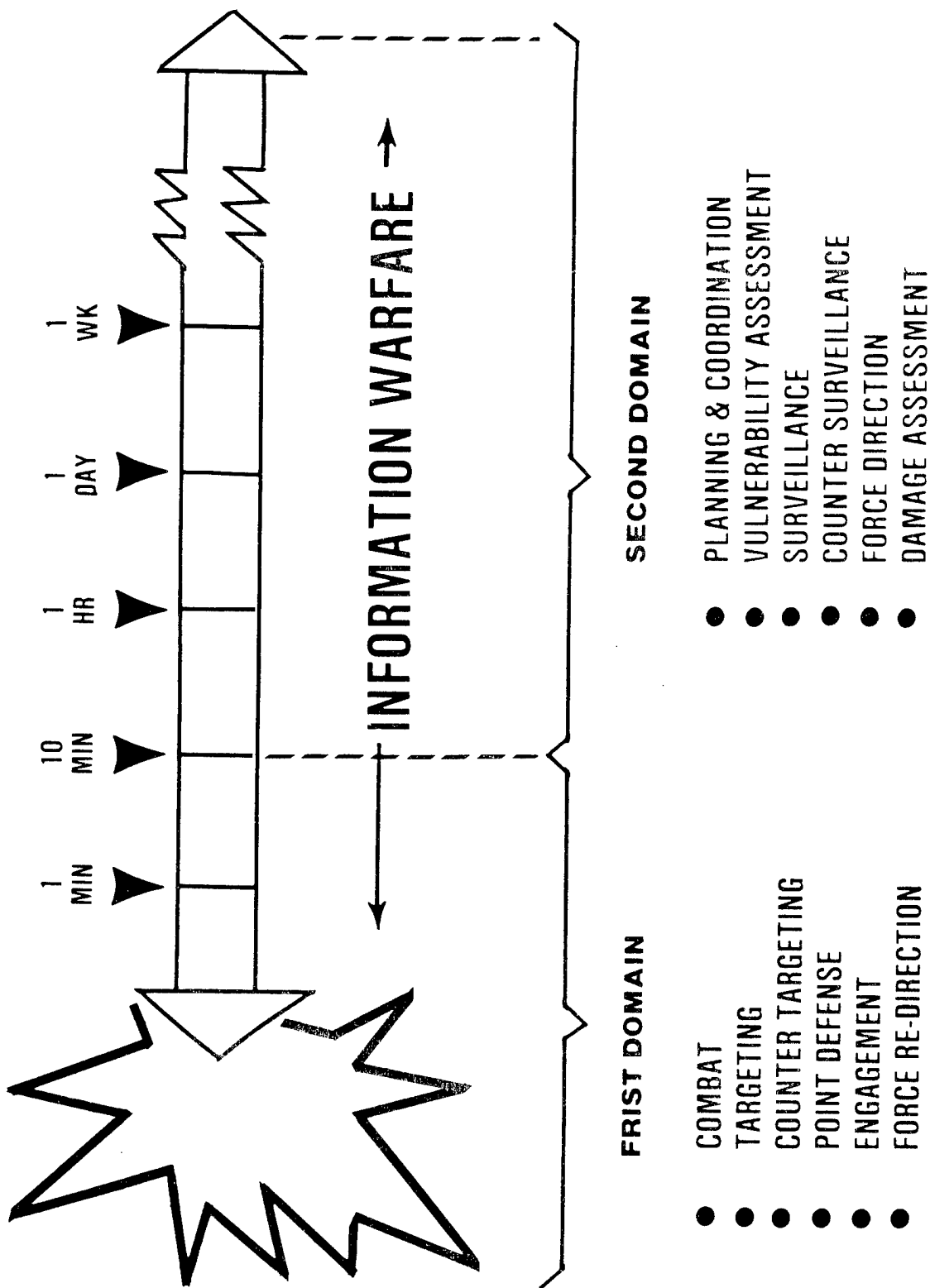


S. PACIFIC

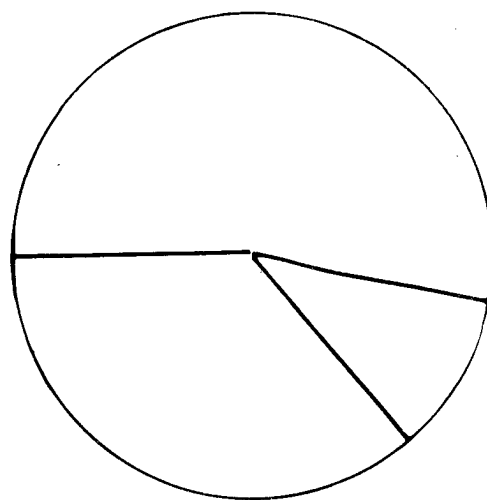
PACIFIC THEATER

- COVERS OVER 1/2 WORLD'S SURFACE
- 60 COUNTRIES
- NEARLY 60% OF WORLD'S POPULATION
- WIDELY DIFFERENT STRATEGIC VALUES
- REGION ACCOUNTS FOR 1/3 U.S. FOREIGN TRADE
- 7 OF 8 LARGEST MILITARY FORCES
- 6 OF 9 U.S. SECURITY AGREEMENTS

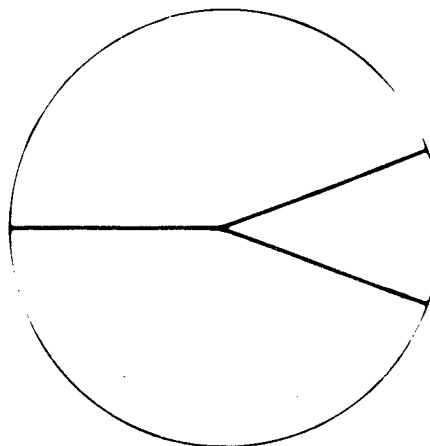
DECISION DOMAIN TIME-LINE



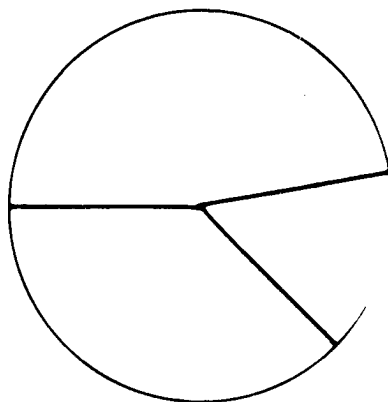
COLLECTION



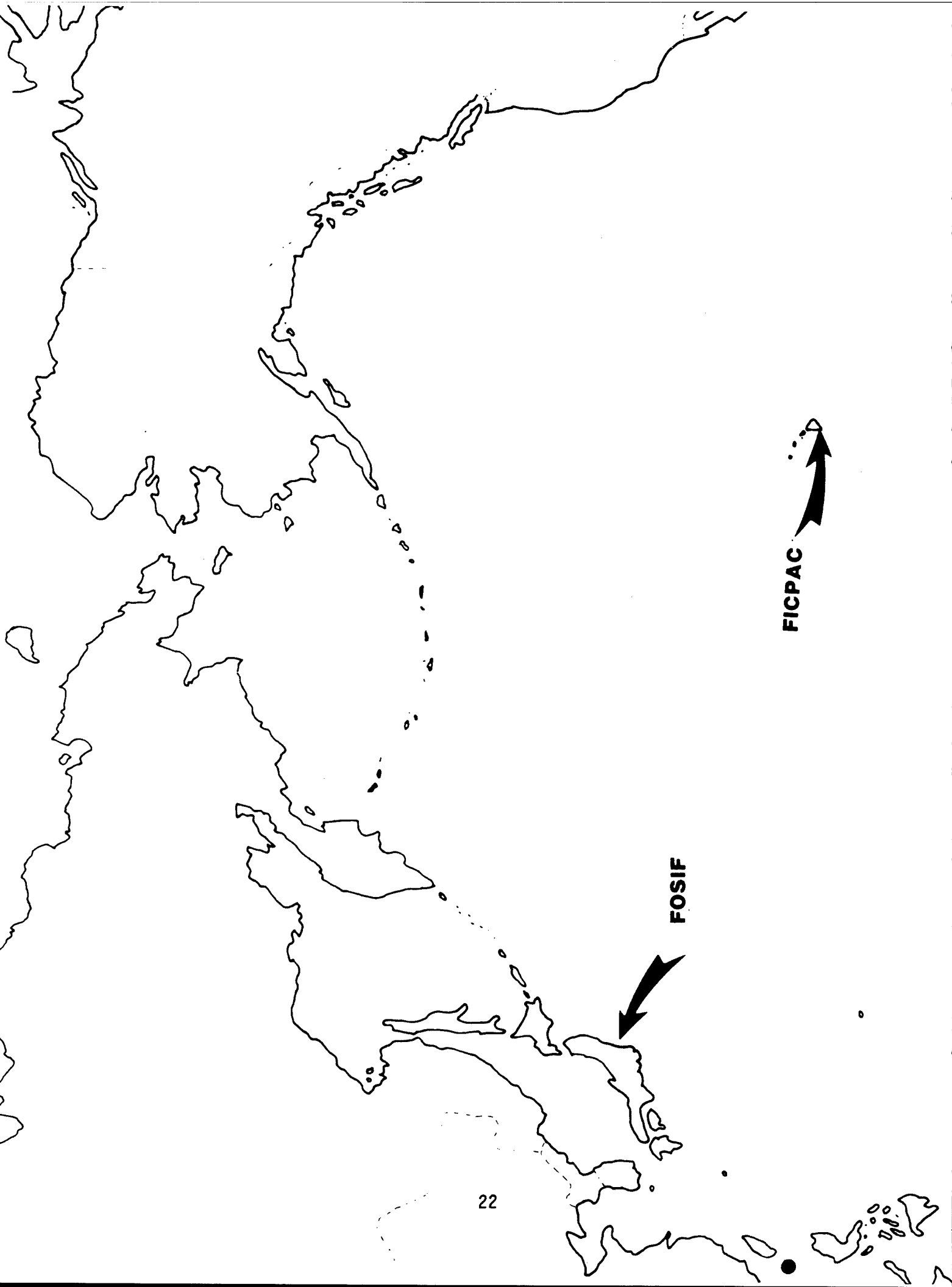
NORTH KOREA

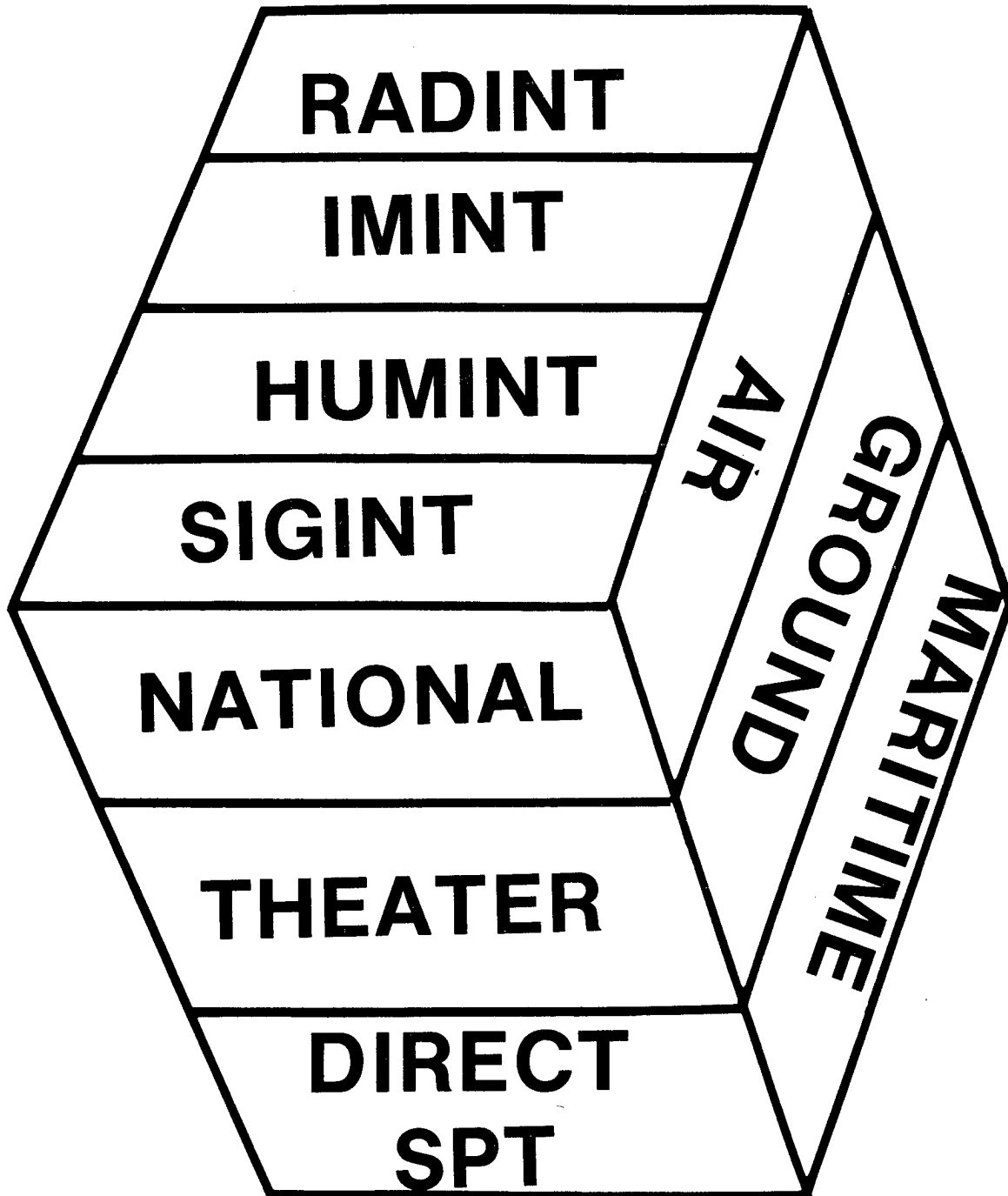


**USSR
(FAR EAST)**



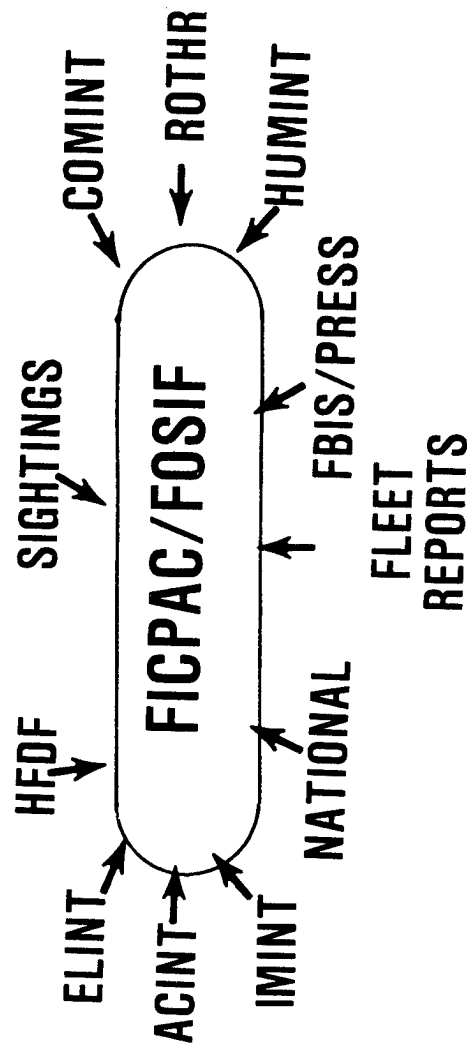
PRC





PLANNING DOMAIN FUSION

● ALL SOURCE IN NATURE



WARFIGHTING REQUIREMENTS SCENARIO

**DECISION
DOMAIN
GEOGRAPHIC
AREAS**

PLANNING
DOMAIN

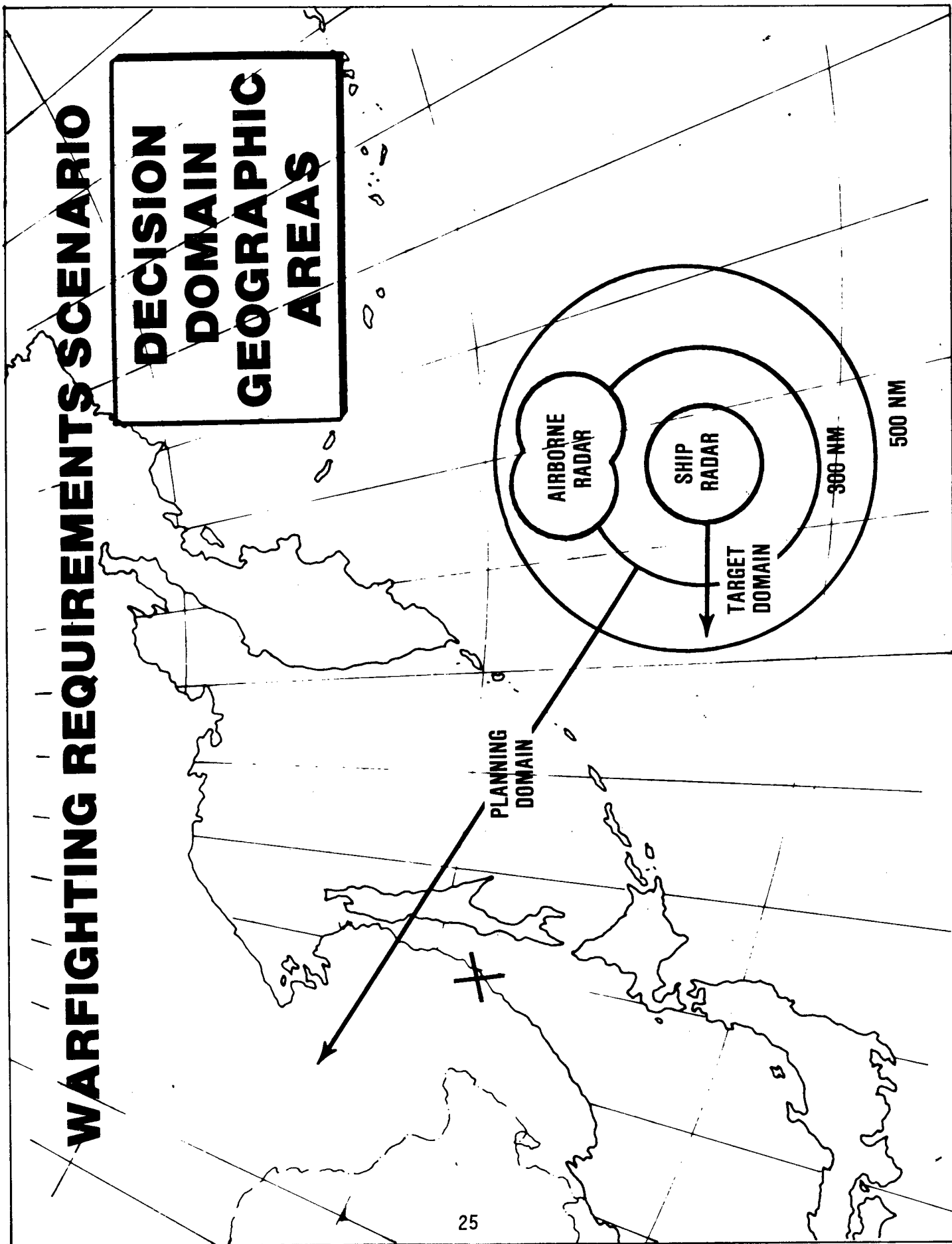
AIRBORNE
RADAR

SHIP
RADAR

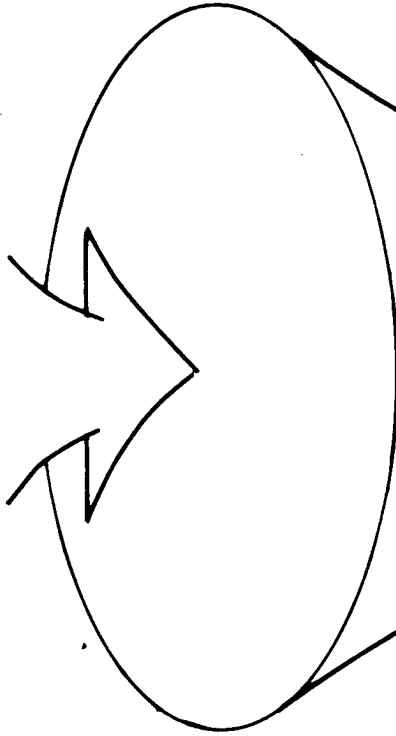
TARGET
DOMAIN

300 NM

500 NM



ALL SOURCE INPUT



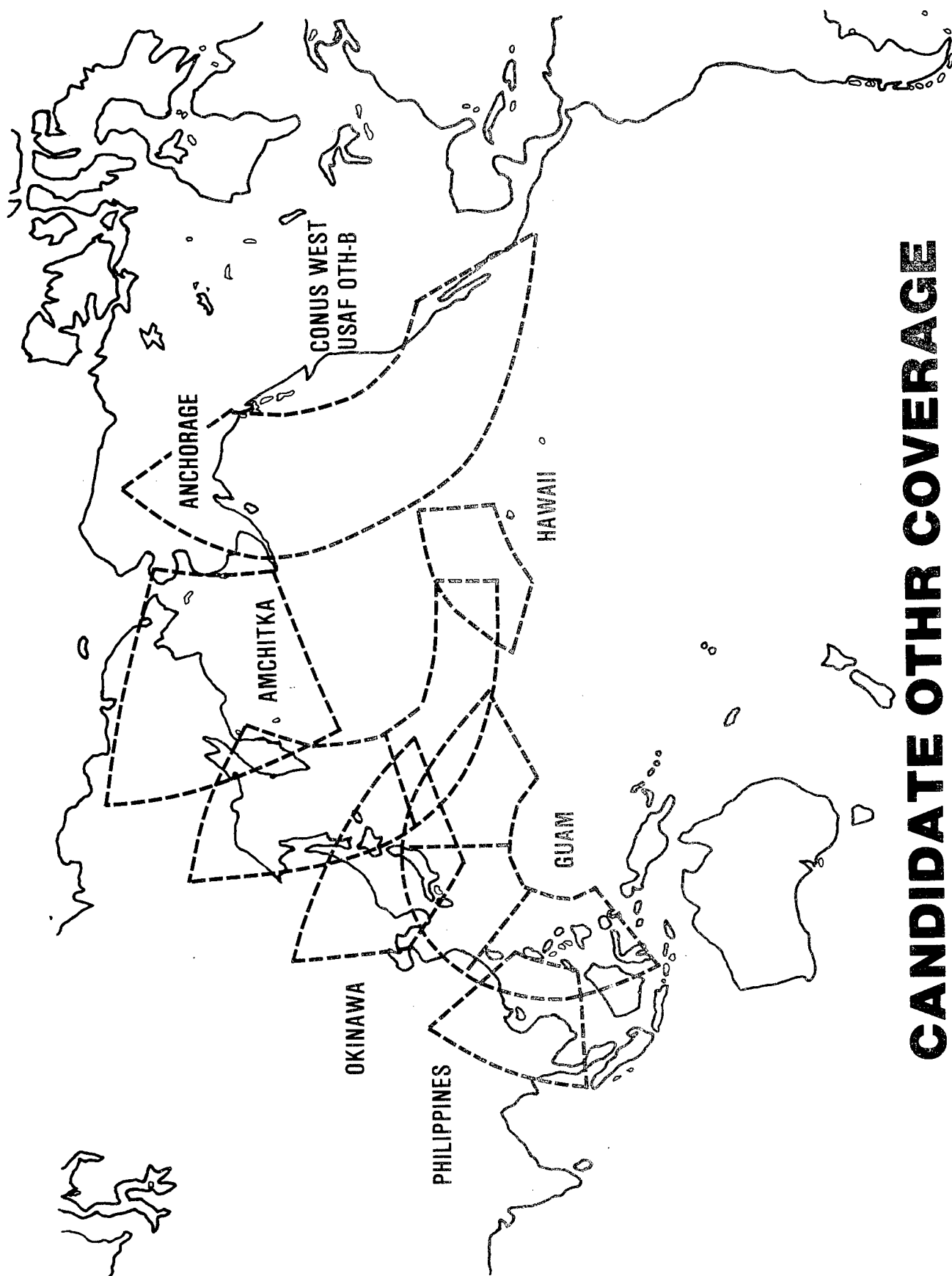
FUSION ASHORE

**FLEET INTEL CENTER
PACIFIC
(FICPAC)**

**FLEET OPERATIONAL
SURVEILLANCE
INFORMATION
FACILITY
(FOSIF)**

**TAILORED FLEET INTELLIGENCE
SUPPORT PRODUCT**

FUSION ASHORE



CANDIDATE OTHR COVERAGE PACOM

NEW COLLECTION SYSTEMS

- ROTH R
- NEW/UPGRADED ELINT SATS
- NEW/UPGRADED PHOTO SATS
- DDS III

FUSION ASHORE

IMPACT OF NEW SYSTEMS

CONTACT REPORTS PER DAY IMAGES PER DAY

	<u>NORM PEAK</u>		<u>NORM PEAK</u>	
1988	2535	10989	400	500
1992	5354	24745	2000	3000

COMMANDER'S CONCERNS

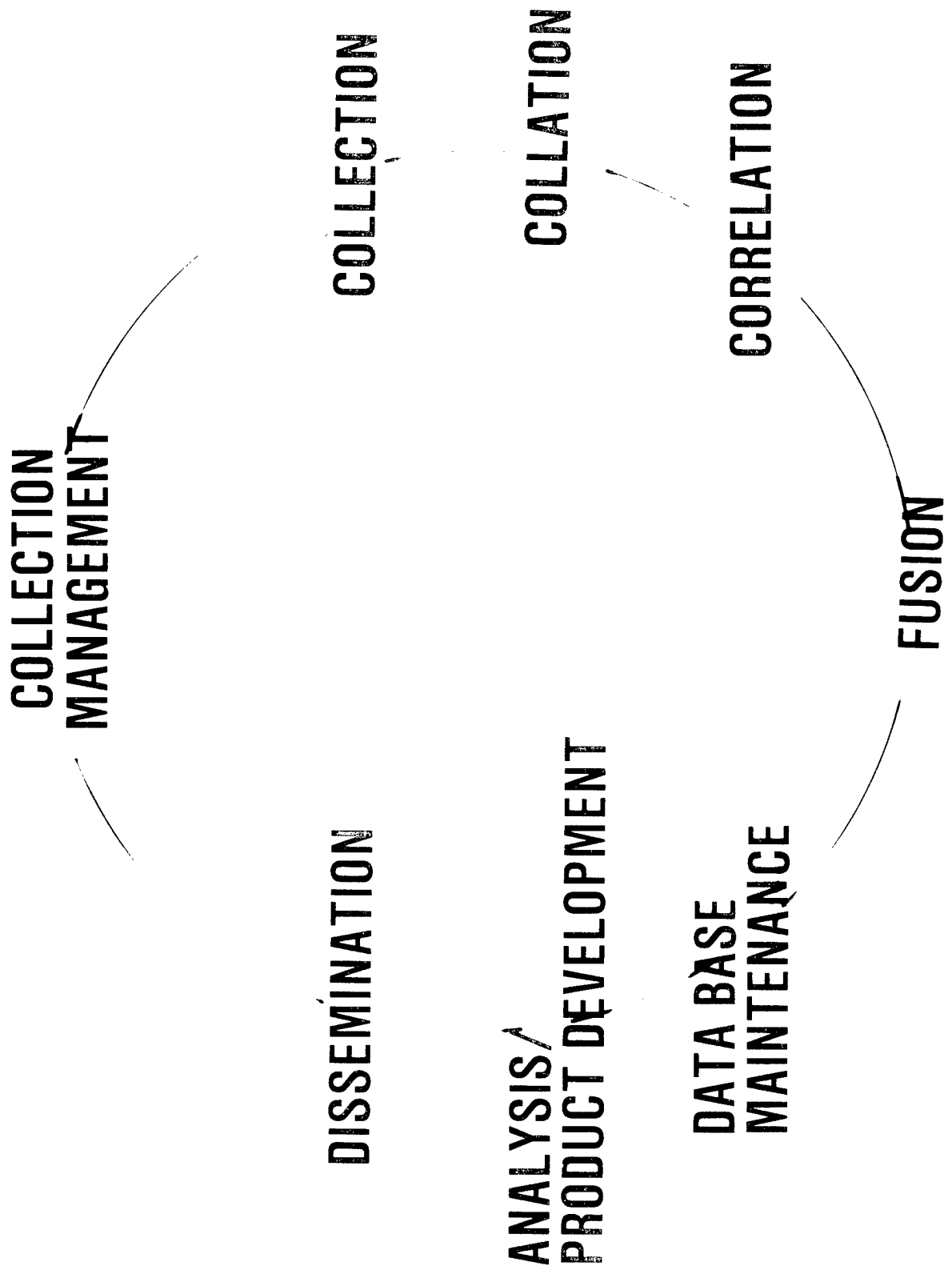
- WARTIME SURVIVABILITY
- COMMUNICATIONS
- TIMELINESS

TARGET DOMAIN SUPPORT

—DIRECT REPORTING TO SHIPS/USMC

**FLEET BROADCAST
DIRECT LINK TO NATIONAL AGENCIES
TADIXS B/TRE
ROTHR
JSIPS
INDIGENOUS SENSORS**

INTELLIGENCE CYCLE



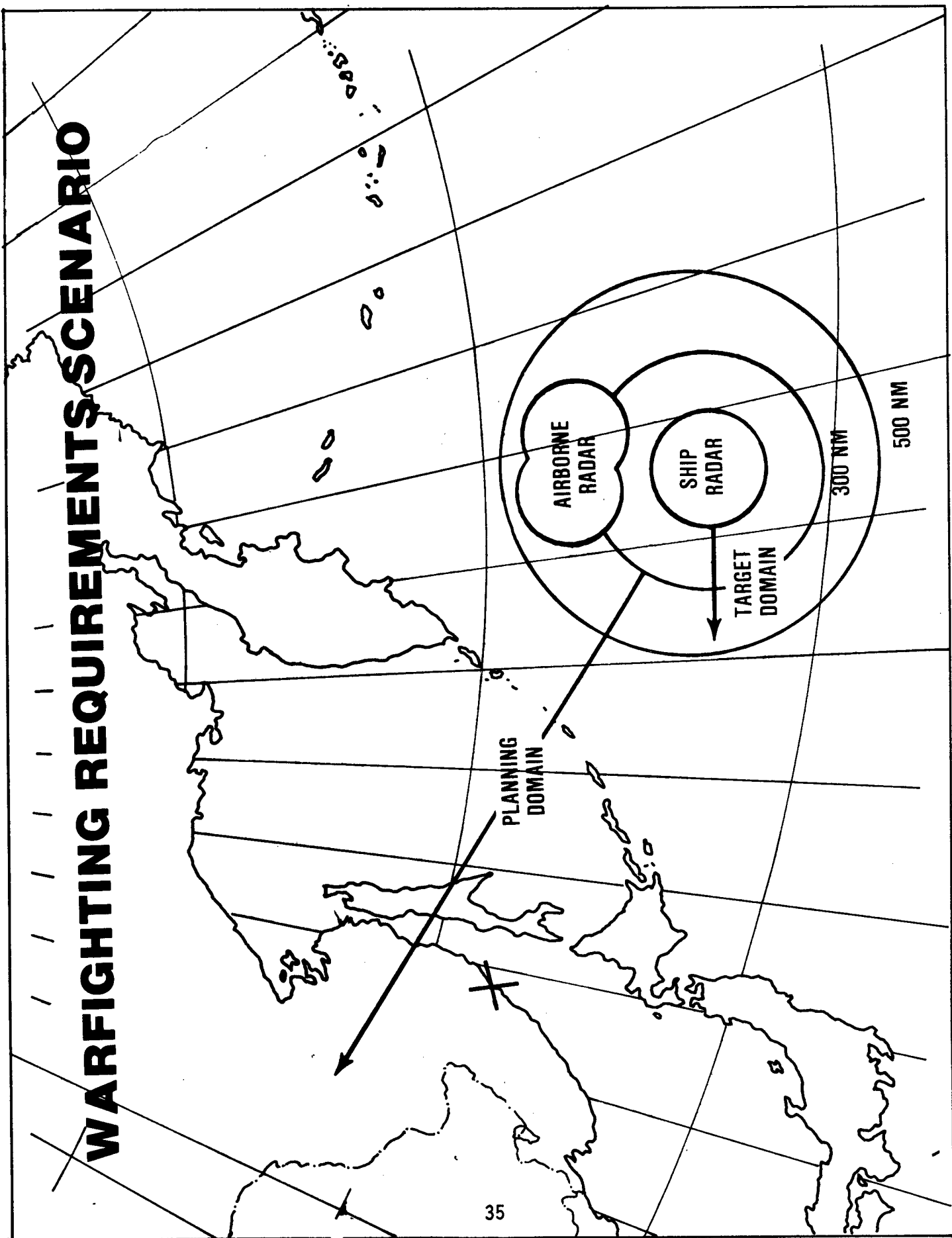
TACTICAL COMMANDER CONCERNS

- DATA OVERLOAD
 - COMM CAPACITY
 - SAME REPORT MANY WAYS
- SYSTEM PROCESSING CAPABILITY
- AFLOAT MANNING/EXPERTISE

WARFIGHTING REQUIREMENTS

- PROPER BLEND BETWEEN AFLOAT AND ASHORE**
- TRULY ALL-SOURCE FUSION CAPABILITY**
- ANALYTICAL TOOLS**
- SUPPORT ENTIRE INTELLIGENCE CYCLE IN SYSTEMS DEVELOPMENT**

WARFIGHTING REQUIREMENTS SCENARIO



THIS PAGE LEFT BLANK INTENTIONALLY

DATA FUSION FOR TACTICAL FORCES

37

1988 Tri-Service Data Fusion Symposium
17 May 1988

Cdr Scott Schneberger, USN
OPNAV 092P22
(202) 694-0297

CDR SCOTT SCHNEBERGER, OFFICE OF NAVAL INTELLIGENCE

The focus of this presentation will be on what I consider to be basic principles of fusion for tactical commanders. In doing so, I will also address two programs that Admiral Studeman initiated within the last two years.

War is chaos! The good news is that it's real hard for the Soviets to figure out what we're doing in the heat of battle; the bad news is that we can't often figure out what we're doing either. My point is that the opposite of fusion is confusion--we can vastly improve tactical warfare by lessening confusion while improving fusion. To that end, I have come up with eleven, basic "fusion factors."

The ultimate recipient of fusion is a person. Unless warfare is completely automated, there will always be a decision-making person between the sensors and the weapon to be delivered on the target. Fusion must serve that person and not just the weapons. No matter how good the sensors are, no matter how good the communications are, no matter how quickly the data comes to the user, and no matter how good the processing algorithms are--if the end result is a decision maker who is confused, then the whole system has failed and warfare is left to instinct or chance.

More and faster is not always better; better is always better. Keeping in mind the person at the end of the fusion train and that this person (particularly in the heat of battle) can only handle so much information, giving more data to a person already confused may not clarify the situation. Indeed, it may only worsen his decision-making. A key requirement that has been brought up a number of times this morning is for clarification--better data--rather than more or faster data. One of the ways to clarify is to tailor the tactical picture.

Tactical fusion must be tailored. The right user must get the right information in the right amount at the right time; anything else increases confusion. The system providing that information must be flexible in order to adjust the tailoring to the situation at hand. If possible, that tailoring should be done by the decision maker; if not possible in the heat of battle, it must be done by some operational intelligence organization "fronting" for the tactical commander. The Intelligence Support for Strike/Amphibious Forces (ISS/AF) effort began for this very reason. ISS/AF, a joint OPNAV and CMC effort led by Admiral Studeman, is an effort to improve our ability to provide support at existing shore-based and afloat sites and provide new tools for those going ashore. The basic ISS/AF problem is that the tactical commander knows his tactical information needs best, but has the least capability to coordinate tactical, theater and national sensors. He has the least capability to process the best amount of data available to him, and even if he could, he has the least capability to transform the information into a product his weaponry needs. ISS/AF seeks to provide those capabilities in a tiered fashion to tactical forces ashore.

Tactical fusion must include mapping, charting and geodesy (MC&G) and environmental data. ISS/AF has highlighted this requirement. All forces, even those in space, fight relative to the earth. Accordingly, tactical information must be presented to tactical decision makers in the context of the earth and its features. Obviously, airplanes can't fly through mountains and tanks don't normally cross lakes. These factors, combined with environmental data and the efforts of the environment must be presented to tactical commanders. For example, has the weather been cold enough long enough for the lake to have frozen sufficiently for tanks and trucks to drive across?

Tactical fusion must include "finished intelligence." Tactical decisions must include reference data such as order of battle, installation data and weapons characteristics. It is not enough to say to a tactical commander "you have two Boghammers closing rapidly at 030." To make a balanced, cohesive tactical decision, he needs to hear something like "there are two out of their four Boghammers, armed with eight five-inch rockets, closing rapidly at 030." The commander needs to have the situation presented to him in the context of the whole threat.

Tactical fusion must allow for multi-level secure data dissemination. The point here is simply that there are many sensitive sources with data that could significantly lessen confusion in the decision maker's thought--data not normally shared with some levels of command due into the system and down to those lower access levels when it's needed and in a timely manner? How do we give him confidence in the data without revealing sensitive sources?

Tactical fusion must be coordinated end-to-end, from the sensor to the weapon. The point here is that fusion is not limited to a processing system or a series of systems; it occurs all along the way from a smart sensor to a decision maker to a smart weapon and it must be coordinated accordingly. We can have tons of high quality sensor data wonderfully fused and sent to the decision maker--but if it all goes through a 75 baud circuit which is backlogged six hours, the system has failed. This coordination must include smart reporting, protocols, communications circuits, system interfaces, data base structures, and man-machine interfaces. A basis for this coordination must include smart reporting, protocols, communications circuits, system interfaces, data base structures, and man-machine interfaces. A basis for this coordination is Information Management or INFOMAT initiated by Admiral Studeman to improve end-to-end coordination in conjunction with sensor managers, command and control entities, intelligence centers, and end users. We cannot afford to fight in the dark or pay for sophisticated sensors that produce unused data.

If confusion is lessened by commonality, watch out for exploitable loopholes. This fusion factor is really a warning. We are hearing more and more about common hardware, common software, common algorithms, common data bases, etc., usually promoted for guaranteed interoperability or lowered life cycle maintenance costs. It's important to keep in mind that commonality can be a two-edged sword; if we have complete commonality and there's at least one exploitable weak node that is successfully deceived, then we will all be deceived and all go over the cliff together.

Decision makers must not only understand the situation, they must believe it. This cannot be overstated. Remember the person in the link from the sensors to the weapons; no matter how good the data, no matter how well it's presented to be understood--if it's not believed, it's worthless. We must not try to dazzle or impress the person at the other end, we must educate him. This is particularly true today when we have so much data automatically processed and displayed; the mental "distance" between the person and the raw data is approaching infinity as we are forced to trust automated processing of information we never see based on software we don't understand. There must be a way, no matter how complicated the fusion system, for the user to say to the system "prove it; show me how you came up with that conclusion."

Knowing intentions reduces confusion. Obviously, a decision maker's job is much easier if he knows the future--the enemy's intentions. We were very successful in conducting antisubmarine warfare in the Atlantic in World War II not because we knew where the German U-boats were every minute of the day but by reading their "mail" and knowing where they were going to be and when they were going to be there. And of course, even if we could have tracked them constantly, correctly knowing intentions is better than extrapolating historic tracking data for putting weapons on a target. The goal is to be able to read a target--to predict through fusion and analysis and through "reading his mail." However, this too is a two-edged sword, as people do change their tactics and intentions. Intentions must be fused with current locational data and capabilities.

Decision makers don't live in the past; their decisions are about the future. The last "fusion factor." It is important to remember, as Admiral Studeman pointed out, that intelligence officers are not merely historians. Their purpose is not solely to catalog the past down to the second and tell us what happened. Their job is to predict the future so that the decision maker can make a decision about the future--not about the past. The fusion system must not lose sight of that. The tactical commander must initiate defensive measures before a threat is imminent and he must lead his target offensively. The fusion process from end to end must allow and even promote future extrapolation.

That's all. But, for purposes of this symposium, it's important to remember these basic fusion principles while discussing complex and state-of-the-art ways to fuse data and improve systems processing. They were identified from the vantage point of the tactical user--the tactical decision maker for whom we build fusion systems.

"War is chaos. The reason Americans
do so well in combat is that they
practice chaos on a daily basis."

--WWII German General

FUSION FACTOR #1

The ultimate recipient
of fusion is a PERSON.

FUSION FACTOR #2

More and faster is not
always better; BETTER
is always better.

FUSION FACTOR #3

Tactical fusion
must be TAILORED.

- right user
- right type
- right amount
- right time

INTELLIGENCE SUPPORT TO STRIKE/AMPHIBIOUS FORCES (ISS/AF)

Joint OPNAV/CMC effort
Rapid capability improvement
to existing ISS/AF sites
Initiated FY87; IOC FY88
6+10 shore-based sites
24 afloat sites
Ashore suites (man portable)

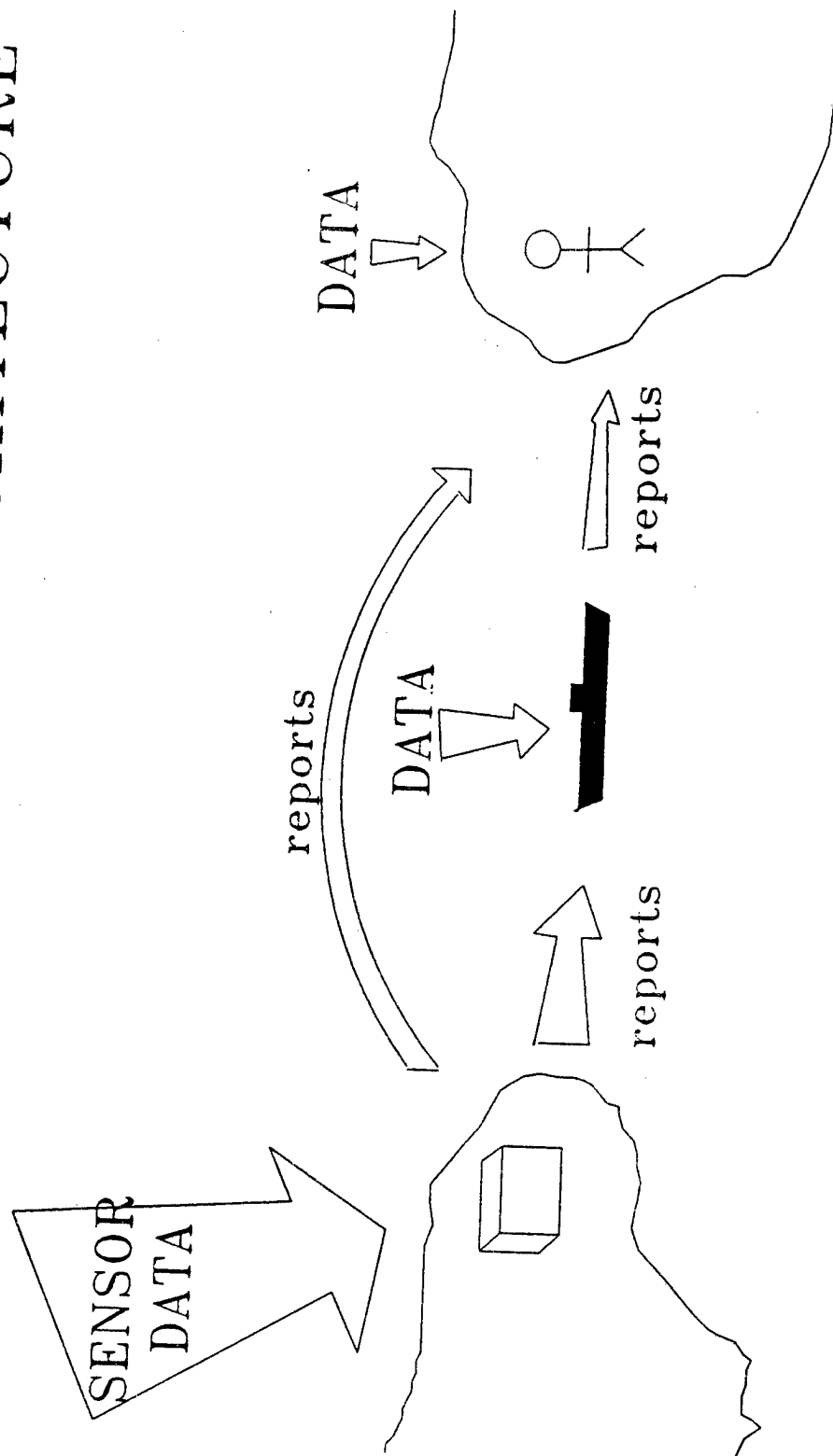
BASIC ISS/AF PROBLEM:

TACTICAL COMMANDER BEST
KNOWS TACTICAL INTEL NEEDS

BUT

- NATIONAL/THEATER/TACTICAL SENSOR MIX IS TOO COMPLICATED TO MANAGE ON SCENE
- DATA AVAILABLE IS MORE THAN TACTICAL COMMANDER CAN HANDLE
- ANALYSIS/PRODUCTION TOO COMPLEX FOR TACTICAL COMMANDER TO HANDLE

ISS/AF CONCEPT ARCHITECTURE



FUSION FACTOR #4

Tactical fusion
must include:

- MC&G data
- environmental data

FUSION FACTOR #5

Tactical fusion
must include "finished"
intelligence.

FUSION FACTOR #6

Tactical fusion
must allow for multi-level
secure data dissemination.

FUSION FACTOR #7

Tactical fusion must
be coordinated top to
bottom--sensor to decider...

FUSION COORDINATION:

- direct, timely reporting
- smart reporting
- data base commonality
- processing
- MMI
- displays
- tailoring
- communications

OPINTEL INFORMATION MANAGEMENT (INFOMAT)

ONI initiative

Coordinated with OTH-T INFOMAT-like
efforts

Sensor to shooter OPINTEL coordination
Flow, content, timeliness, accuracy
DNI leads/chairs

OPNAV/ONI/SPAWAR Guidance Committee
Field technical reps assigned
to FltCinC N2s

FUSION FACTOR #8

If confusion is lessened
by commonality, watch out
for exploitable loopholes!

FUSION FACTOR #9

Decision maker must not
only understand situation
presented, he must
BELIEVE it.

FUSION FACTOR #10

Knowing intentions
reduces confusion.

FUSION FACTOR #11

Decision makers don't
live in the past;
their decisions are
about the future.

**REPORT OF THE
DEFENSE SCIENCE BOARD TASK FORCE ON**

**COMMAND AND CONTROL SYSTEMS
MANAGEMENT**

July 1987

**Dr. Solomn J. Buchsbaum, Chairman
Vice President, Bell Laboratories**

PRESENTED BY FRANK WHITE, NOSC

1978 BUCHSBAUM TASK FORCE REPORT:

"The nation is failing to deploy command and control systems commensurate with the nature of likely future warfare, with modern weapons systems, or without available technology or industrial base."

1987 BUCHSBAUM TASK FORCE REPORT:

"We conclude that, indeed, outstanding progress has been made in the past decade, especially in strategic command and control. The present command and control infrastructure is more extensive as well as more resilient than that of a decade ago."

DEFENSE SCIENCE BOARD TASK FORCE APPROACH

"Our emphasis is on the commander—the commander at any level—helping the commander acquire and evolve the command and control system that fits his needs. This approach is, of course, fundamentally different from the standard acquisition process wherein evolution is not the central driver and in which the commander's involvement is limited to participation in the establishment of requirements."

"The time is propitious to build further upon the progress already made during the past decade in command and control systems management and the ongoing reorganization of the Department of Defense."

TASK FORCE DEFINITION OF IDEAL COMMAND AND CONTROL

"... a command and control system supporting a commander is not just communication links; and it is not just all the information processing and fusion that must go in any well-designed and well-operating command and control system. It is all of the above and much more. The ideal command and control system supporting a commander is such that the commander knows what goes on, that he receives what is intended for him, and that what he transmits is delivered to the intended addressee so that the command decisions are made with confidence and are based on information that is complete, true, and up-to-date. The purpose of a command and control systems is, in the end, to provide assurance that orders are received as originally intended with follow-up in a timely fashion, which can make the difference between winning and losing wars."

MAJOR AREAS OF COMMAND AND CONTROL SYSTEM SHORTFALLS

- There is still a gap between the command and control systems we should be fielding and those that we are fielding.
- The process of determining and acting on command and control needs is not working as effectively and speedily as it must. The acquisition process needs strengthening. There are tight resources. There is the lack of an agreed-upon, well understood DoD architectural framework with its well-defined interfaces and standards to guide the evolution of command and control systems. There is a dearth of personnel skilled in command and control.
- There is too much emphasis on the development of completely new systems over the evolution of the very comprehensive already in-place systems.

MAJOR AREAS OF COMMAND AND CONTROL SYSTEM SHORTFALLS *(Continued)*

- There needs to be an established integrated command and control support activity with responsibility for architecture, interoperability standards, and technical support activities with responsibility for support for all aspects of command and control including exercises, doctrine, and training.
- The greatest near-term national improvement in command and control is possible through increased effort at the vital lower level of interface.
- There is a need to greatly improve the operational performance and readiness of current command and control systems through test, fix, and test again evaluation exercises.

MAJOR AREAS OF COMMAND AND CONTROL SYSTEM SHORTFALLS (Continued)

- User oriented testbeds are required which are closely coupled to the acquisition process are needed to maximize the operational effectiveness and utility of programs to upgrade command and control.
- Operational testing and exercising at all levels of the command structure worldwide should be fostered and strengthened throughout the life cycle of the command and control system life cycle.
- If the CINCs are to play an appropriate role in command and control management, they must have access of the necessary architectural and technical support and the funds the role requires.

MAJOR AREAS OF COMMAND AND CONTROL SYSTEM SHORTFALLS (Continued)

- There must be an institutionalization within DoD of a process for incremental evolutionary acquisition of command-unique systems under CINC management. This must be accompanied with the resources necessary to carry out these responsibilities.
- The command and control research is unfocused and largely technology oriented. This research does not address the important and challenging operational problems command and control faces.
- We must be willing to develop the research specialists necessary to ensure that we are at the leading edge of command and control technology.

MAJOR AREAS OF COMMAND AND CONTROL SYSTEM SHORTFALLS *(Continued)*

- Embedded training would permit those involved in command and control activities to conduct training in the most realistic environment possible and would permit the command and control users to provide feedback to the command and control developers that would aid in ensuring that our command and control systems are responsive to the changing needs of our commanders.

RECOMMENDATION AREAS

- 1. Assuring the operational effectiveness of systems for command and control.**
- 2. Strengthening the capabilities for testing and exercising as well as evaluating and specifying functions performed by command and control systems.**
- 3. Strengthening the capabilities of the commands to upgrade and evolve their command-unique systems.**
- 4. Improving the regulation for the acquisition of command and control systems.**

RECOMMENDATION AREAS

(Continued)

5. Strengthening the intellectual base for command and control through a coordinated program of research in command and control.
6. Strengthening education and training as well as career pathing of command and control professionals.

RECOMMENDATION AREAS

1. Assuring operational effectiveness

- Establish and maintain architecture for the command and control of US forces operating under either national or allied command that links all elements of the command and control structure from both top-down and bottom-up.
- Establish and maintain the standards needed to achieve interoperability and operational effectiveness in the field and enforce adherence thereto.
- Provide conceptual guidance and technical support to field commands as they evolve their command and control systems within the overall architecture and interoperability standards.
- Identify and approve Required Operational Capabilities (ROCs) that are timely and responsive to the inputs and needs of the CINCs.

RECOMMENDATION AREAS

2. Specifying, Testing, and Exercising

- The capabilities of the JCS, CINCs, and Services for operationally testing, exercising, evaluations, and specifying functions to be performed by command and control systems should be fostered and increased throughout the life cycle of the command and control systems

RECOMMENDATION AREAS

3. Upgrade and Evolve Systems

- Each CINCs involvement in the planning, funding, and acquisition of command and control systems relevant to his command be increased and that each CINC have resources organic to that command to evolve, upgrade, and maintain his own command-unique command and control system under the overarching architecture established centrally by DoD. The DoD should institutionalize this process of incremental evolutionary acquisition of command-unique systems under CINC management and with Service support of the required technical infrastructure.
- Each CINC should have access to his own small architecture capability to help provide the information needed for interoperability, for inputs to the CJCS on JROCs and priorities for command and control funding and for design of the Command's unique command and control needs.
- A modest increase in CINC initiative funds be made to provide the means to meet time-sensitive command and control needs of the CINC.

RECOMMENDATION AREAS

4. Regulation for Acquisition

- The DoD should issue new directives to govern the acquisition of command and control systems that recognize the special characteristic of those systems. These directives should recognize that the various stages of the development of command and control systems overlap; recognize that users participation in the conception, evolution, testing, and development of command and control systems is a strong requirement; and provide flexibility and adaptability to meet the wide variations in the needs of commands.
- Urge that the acquisition policies and oversight processes for command and control now being developed take into account the thrust of this recommendation.
- Recommend that DoD regulations reinforce and enlarge the guidelines contained in the Defense Acquisition Circulars and that compliance with such guidelines be assured.

RECOMMENDATION AREAS

5. Strengthening the Intellectual Base Through Research

- A comprehensive program devoted to research on command and control should be defined and implemented. The research program should delve into all aspects of command and control—not just the technological aspects. It should form close linkages to the several research and graduate education programs in command and control in Service and Defense educational institutions as well as exploit and foster related research programs in our universities.

RECOMMENDATION AREAS

6. Strengthening Education and Training

- The Director for command and control systems in the Joint Staff should identify the requirements for command and control specialists in the Services, the Joint Staffs, the unified and Specified Commands, and develop with the Services the internal Service manpower requirements to include those needed to ensure viable R&D programs.
- The Services should develop command and control career patterns that ensure adequate personnel are assigned to as well as developed in the command and control speciality in a career progression that provides adequate incentives for their continued service in this speciality.
- The JCS and the Service should develop an educational and training system that fully supports the command and control needs of the military forces.
- Embedded training and programs should be developed to enable training to proceed in the most realistic environments as well as to provide improved feedback to developers.

UNDER SECRETARY OF DEFENSE RESPONSE

- Assistant Secretary of Defense (C3I) will coordinate action responsibility assigned to:

**OJCS
DARPA
DUSD (T&E)
DCA/JTC3A
ASC (C3I)**

JOINT DIRECTORS OF LABORATORIES (JDL) required to respond to DSB recommendation-

- To Strengthen the Intellectual Base for Command and Control**

Generated a Five-Year Plan (FY90-94) to accomplish these objectives

THIS PAGE LEFT BLANK INTENTIONALLY

DATA FUSION RESEARCH MANAGEMENT

DR. DANIEL F. WIENER II

UNIVERSITY OF VIRGINIA
SCHOOL OF ENGINEERING AND APPLIED SCIENCE

INTRODUCTION

The fusion of data to provide information is not a new endeavor. It has been present throughout recorded time. Heretofore, it was always accomplished by a living, thinking organism.

War fighters, in particular, have long recognized the need for information, not data, but have had a difficult time in describing the totality of their requirement. With the proliferation of electronics-based sensors, the streams of data have gone from light mists to raging torrents eclipsing the capability of manual approaches. As such, the information management community has started down several grand and glorious paths to use electronics to automatically solve the "problem" automatically, i.e., deriving information from data. While one might be expected to state that the problem, as well as the solution, is routed in technology, it is this paper's premise that the end can only be accomplished through a judicious use based on a realistic research approach with sound research management.

BACKGROUND

Data fusion research management has been thought of as an oxymoron. Data fusion research and data fusion management appear to be conflicting concepts. It was not conceived to be that way but evolved because of what is commonly called the "Quest-For-Research-Gold."

The Quest-For-Research-Gold concept can be described as, "if there is funding for a research area, I want to get my fair share." Fair share is defined by, and is in the eyes of, the beholder. Like Artificial Intelligence a few years past and Artificial Neural Systems today, data fusion is a "hot" topical research area. Therefore, everyone within the government research community wants, and, I might add, can justify, a piece of this research dollar pie. Furthermore, no consultant, industrial organization, or academic institution wants to be left out. The result is research chaos. However, as

Nietzsche said, "Out of chaos comes order." With the emergence of Joint Directors of Laboratories - Data Fusion Subpanel (JDL-DFS) and the Joint Tactical Fusion Program (JTFF), as well as international organizations like the SHAPE Technical Center Research Panel, order is in sight.

DEFINITION

The history of the data fusion research is fraught with many long-standing arguments based on semantics. JTFF and JDL-DFS have sought to resolve this issue with the drafting of candidate definitions. For the purpose of this discussion, the following JTFF definitions are used:

Analysis: A series of technical/cognitive actions taken to convert collected data into reports.

Correlation: The process that combines and associates reports/observables from the same or different sources that refer to the same battlefield entity, to form a single record that consists of the best available information on that entity.

Aggregation: The process of combining/associating reports/observables, to identify parent organizations/subordination based on inferences drawn from the presence of its component parts.

Fusion: The function of providing an Intelligence Product from merged reports based on multiple sources and systems from multiple Intelligence disciplines that supports the situation analysis and user comprehension of battlefield dynamics.

While these definitions may not be perfect, they were derived through an Adelphi process with the military and technical leadership in 1983 under Joint Tactical Fusion Program sponsorship. Although longer than necessary, they have served the test of time during a period of increased awareness in data fusion tech-

nologies and systems. No such definition exists for all source data fusion. One man's all source data fusion is another man's multi-source. The issue of definitions is further complicated when one considers the use of preprocessors -- not withstanding ownership of data as issues. However, the important issue is to establish a basis of context for discussions.

PREJUDICES

The prejudices underlying this article are based on almost two decades of work in tactical warfare technology, particularly concentrating on interdiction/strike weaponry and most recently as the Deputy Program Manager and Technical Director of the Joint Tactical Fusion Program.

Initial involvement with what was described as fusion was with the U.S. Navy during Vietnam. VQ squadron support was SIGINT-based with acoustic augmentation to provide a situation assessment of the enemy, as well as real-time missile and enemy aircraft warnings to our strike aircraft. The shrouds of secrecy prevented widespread dissemination at the time. Fusion was defined by RADM Roy M. Isaman (deceased) as "what you do to get useful information from data -- but in many different forms." The same data would be "fused" in many different ways with many different sources to provide as many different products.

Subsequent activity also involved the use of heretofore "green door" products to further tactical warfare and tactical weaponry. One of these was the Cruise missile, another, more grandiose concept was Assault Breaker. The original Assault Breaker concept was based on three elements. A large, stand-off airborne radar; a modular warhead, "long-range" missile; and a glue to interface the two. This glue was reminiscent of the discussions on "cosmic glue" used to hold the elements of an atom together. As pi-mesons, mu-mesons and gluons were discovered, real-time data fusion systems were also discovered.

BACKGROUND

To understand the present, we must look to the past, and the past is the Battlefield Exploration and Target Acquisition Program, BETA.

The tri-service BETA Program was established by the Office of the Secretary of Defense in 1977 as a technical demonstration project with an objective to participate in the REFORGER-80 exercise. The purpose was to support battlefield commanders in the execution of the air/land battle, particularly with respect to resource allocation, maneuver of forces, and targeting. The approach of BETA was to correlate/fuse multi-disciplinary sensor products (alphanumeric messages) into a coherent graphic picture of the battlefield with an implementation based on commercial hardware and the development of correlation software to accomplish first-order data fusion that could be used to support

the development of individual service data fusion systems. The BETA project ran into continual problems with interpretation of requirements. As such, BETA was unable to participate in REFORGER. BETA was officially concluded in February 1981 when the the two test beds were accepted at the contractor's plant and subsequently deployed to the Army at Ft. Hood, Texas, and to the Air Force at Hurlburt Field, Florida. These test beds were to be used for user testing at these sites to refine requirements and support the subsequent All Source Analysis System/Enemy Situation Correlation Element (ASAS/ENSCE) development.

A composite capability of BETA fashioned from the contractor's software development facility and some of the Ft. Hood and Hurlburt assets was deployed to Europe in October 1982. This BETA Derived System for the U.S. European Command system, known as Limited Operational Capability Europe (LOCE), was to support both U.S. and NATO forces. In November 1987, the LOCE role was expanded to serve as a gateway to NATO. The Hurlburt test bed was repackaged and deployed to Shaw AFB in 1985 to support the Gallant Knight exercise in February 1986. The exercise was extremely successful, resulting in the Air Force Commander, Lt. General Kirks, commenting that "it was the best intelligence support he had ever received." As a result, the system is in the process of being upgraded to serve as an interim ENSCE capability. A like system has been replicated for the 12th TIS at Bergstrom Air Base.

JTFP was established originally in 1981 as an outgrowth of the old BETA project. Many jurisdictional battles ensued with government agencies, contractors and congressional staffs all having the "correct solution." As a result of a recommendation of the study led by Brigadier General Phillip Mason USA, a new JTFP was established as a Congressionally directed acquisition in 1983 to bring together all military tactical intelligence data fusion development activities. The program has a unique reporting structure. It reports currently to a Joint Oversight Group. This group is comprised of the same members one would see at a Service/Defense Research Board with representation from the Assistant Secretary of Defence for C3I.

Ten major projects under the JTFP are taking place. The keystone project is the All Source Analysis System/Enemy Situation Correlation Element (ASAS//ENSCE). JPL is the major contractor to this joint Army-Air Force program with two names. The remaining projects include the Quick Reaction Multicolor Printer, an effort to put essentially a Xerox machine on the battlefield; the Intelligence Digital Message Terminal, AN/PSC-2; the Litton handheld device used for plugging digital data into the overall system; the Army's Technical Control and Analysis Centers, the first automation support to the Army's CEWI functions at Division and Corps, as well as, the First and Second Radio Battalion for the U.S. Marine Corps; the Digital Topographic Support System (DTSS), managed by the Engineering Topographic Laboratory

to provide digital maps to the battlefield; the Integrated Meteorological System (IMETS), managed by the Atmospheric Sciences Laboratory to provide an integrated weather product; the previously discussed BETA Derived Systems, Limited Operational Capability-Europe, LOCE, located at Ramstein Air Force Base under the operational control of US EUCOM, represents an interface into the Allied Theatre of Operations for multisensory data fusion.

Also under derived systems are the LENSCE, Limited Enemy Situation Correlation Element Systems. These systems provide near term automation support to the intelligence functions at the TACC for the 9th Tactical Intelligence Squadron and the 12th Tactical Intelligence Squadron; the Intelligence Fusion Analysis Technology Validation Project, the 6.3b advanced technology bridge and TACSIM, the tactical simulator. The TACSIM program was placed under the JTFP umbrella because of its use as a driver for testing and training activities for the ASAS/ENSCE program. Its primary purpose, however, is to provide simulation support for intelligence operations at U&S command exercises. To preclude duplication, and insure connectivity, Congress directed that all data fusion activities must be accomplished under the JTFP umbrella. The Army fully embraced this concept by placing under a single umbrella all of the projects necessary to assure an integrated weather enemy and terrain product to the commander.

The Joint Tactical Fusion Program neither had, nor desired to have, a research laboratory. As such, JTFP worked with the services to sponsor the Joint Directors of Laboratories, Data Fusion Subpanel to bring together overlapping research areas. This JDL initiative also takes its lineage from the Herman Report of 1982 which identified 52 data fusion centers within DOD where data fusion or related research is conducted. It is, therefore, obvious that the Data Fusion Subpanel program can act as a central clearing house, or point of focus, for coordinating individual services and agencies data fusion research and development efforts. The Data Fusion Subpanel directorate is now asking questions regarding its relationship to other parts of the Department of Defense establishment, and how it can gain the necessary leverage and visibility for its programs. This may be an important question for bureaucratic purposes, but is not necessary to the well being of their mission. The better question is "how to present a unified coordinated research program."

RESEARCH GOAL

What is the goal of data fusion research? This can best be addressed by looking at the Army All Source Analysis System/Air Force Enemy Situation Correlation Element (ASAS/ENSCE). ASAS/ENSCE is a multi-billion dollar program to automate data fusion. The purpose of the program is to provide timely, accurate, organized information that:

- Facilitates decision processes;
- Accrues maximum benefit from assets;
- Clarifies and adds depth to the commander's view of the battlefield;
- Gives a common view of the air/land battle to the ground and air commanders; and
- Generates an opportunity to seize and retain the initiative.

Thus, the commander is placed in the "driver's seat." That is, to provide the right information at the right time and in the right format. The goal is to give a common picture of the battlefield and really bring intelligence to a service function to the battle commander. However, this is a monumental effort. But like all programs, one must start someplace; one must have intermediate milestones; one must finish. When automating a function that heretofore has been manual, it is necessary to take it in bite-size steps. One cannot eat the whole elephant at once. One must do him one bit at a time, and this one is an elephant.

As a result, ASAS/ENSCE developed a policy and built a program around the concept of evolutionary development, where evolutionary development was defined as:

The process of incremental development and fielding of a system with maximum user participation. The process begins with the development and fielding of a minimum capability baseline system with a lead field user. Feedback based on system performance in an operational environment guides the incremental growth from the baseline to the full objective system capability.

The philosophy/approach is summarized as build a little, test a little, analyze, build a little more, test a little more, analyze, work it through in very close contact with the user. If you were to think of our community as a commercial organization, it is staying in touch with our customer and giving him what he needs.

There are major benefits to this approach: the ability to incorporate advanced technology solutions as they become available, such as lighter, smaller, more powerful computers and AI. AI, in particular, allows for problem definitions, problem solving, and problem understanding in this incremental process and for incorporation of that knowledge into the ASAS/ENSCE. A potential side benefit of this is that the shock on the system, that is, U.S. Army and Air Force operations worldwide, will be minimized by allowing their concepts of operation to evolve with the development of the system.

This philosophy is in conjunction with, the entire multi-million dollar DARPA Machine Intelligence research area, whose goal is to develop fundamentally new capabilities for information processing that make computers intelligent through a process of demonstrating computational equivalents to human cognition, perception, and action. In the data fusion research area, the following three major challenges were defined as:

- Automation of the tactical battlefield;
- Fusion of multi-source data; and
- Dissemination of product.

The implied primary needs are: the integration with deterministic systems and the cooperation with other intelligent systems. The secondary needs are: learning, uncertainty management, and causal reasoning. As such, the magnitude of the dollars has resulted in a virtual stampede to have one's own data fusion research program and system.

CONGRESSIONAL INVOLVEMENT

Congress had a heavy hand in the establishment of the BETA program and its successor, Joint Tactical Fusion. HPSCI, HAC, and SASC were the prime players. Studies by the Congressional Surveys and Investigation staff in the late 1970's found that the fusion of intelligence data into a format having utility for the combat commander was one of the weakest links in our tactical intelligence system. Congress cut the 1978 budget for fusion centers by 10 percent to encourage program consolidation. A subsequent, Congressionally directed study done by the Institute for Defense Analysis showed 52 fusion center projects in progress. BETA, its successor, Joint Tactical Fusion, and the Joint Directorate of Laboratories Subpanel on Data Fusion were natural outgrowths of the Congressional interest and concerns. When the BETA software proved insufficiently stable to support the European Demo in the fall of 1980, Congress directed that BETA be concluded and a Joint Tactical Fusion program be organized with the initial manpower drawn from the BETA program. Joint Tactical Fusion is building ASAS/ENSCE for earliest possible delivery and to capitalize on BETA and related fields and ongoing developments.

This implies that the solution to Data Fusion Research management may not exist, but there are questions and lessons that, if answered and assimilated, can provide a road map. But we must finish with the definition of the problem.

Consider General Hagaman's question of last year's symposium, "Is Data Fusion a discipline or a community?" General Hagaman pointed out that Webster defines community as it relates to professionals as "a body of persons of common and special professional interests scattered through a larger society." The data fusion participants are members of a viable, expanding, maturing community, but to what end? The institutionalization of a data fusion

community and/or formalization of the structure by forming a data fusion professional association does not aid or abet the development and fielding of a data fusion capability. This symposium, as Co-ordination Task 4 of the Data Fusion Subpanel Program, has resulted in an increased awareness of what we can and cannot do while serving as a sounding board to assure cooperation without duplication. As such, the creation of a new body is not required.

But what is the problem? The problem is: how do we manage data fusion research with the goal of achieving near-term fielding of systems. The how is a complex issue with no single correct solution. The lessons learned from the past suggest that central management points are required. Organizations, such as the Joint Tactical Fusion Program and Joint Directors of Laboratories Data Fusion Subpanel, can serve such purposes.

SUMMARY

In summary, the fusion of data into information is a cognitive process older than man. We like to say that man is the "thinking animal," but as we learn more about the thought process we realize that the combining of sensory data into information for the subsequent control of actions can be a multi-leveled activity. We are currently at the level of the slug, which has caught the fancy of many researchers in Artificial Neural Systems. The slug cannot be placed on the same level as a dog, let alone man. However, it does have sensors which input data into a multi-source fusion process to control its actions and reactions. The lesson we must learn is this is not a simple problem that will be solved overnight. All source, multilevel data fusion is a goal -- achievable but not in the near term. The triumvirate of government, industry, and academia must not over-require, over-specify, over-commit, or over-sell. That is not to say that we cannot push the state-of-the-art aggressively. However, only through honesty and joint efforts will the desperately overarching goal be achieved.

NAVAL COMMAND AND CONTROL SYSTEM - AFLOAT

NCCS(A)

The Composite Warfare Commander's
Tactical Decision Support System

Dr. A.N. Hafner, Lt. Comdr. C.T. Sutherlin, and Capt. B.A. Thompson

NCCS Afloat Program Management Office (PMW162)
Commander, Naval Warfare Systems Command, NC-1
Washington, D.C., 20363-5100
Attn: Lt. Comdr. C.T. Sutherlin PMW162-222, Rm. 8E08.
(202) 692-4547

ABSTRACT

This paper describes the requirements of the recently approved NCCS(A) Program based on an appreciation of its future operational environment. It stresses Decision Support applications to the data correlation requirements of the system as it is currently planned for installation on CV/CVN and LCC class ships.

The NCCS(A) consolidates all of the processes required to support the historically segregated and largely manual functions of warfare assessment, planning, and control into a single architecture. Advanced techniques for data correlation, fusion and storage; and of Information management, analysis, retrieval, sharing, and display will be incorporated into the system to facilitate decision making by the Battle Force/Group (BF/BG) Commander and his staff.

The paper describes the incremental integration of standardized components (such as the advanced color workstation) into a group decision support complex within the Tactical Flag Command Center (TFCC). It presents the integrated data fusion and spectrum management components of the NCCS(A) system. A top level discussion of the system architecture is followed by a detailed description of each of its information management components. Particular attention is paid to the dual security bus that connects its distributed components throughout the command and control spaces of the platform.

NAVAL COMMAND AND CONTROL SYSTEM - AFLOAT

NCCS(A)

The Composite Warfare Commander's
Tactical Decision Support System

Dr. A.N. Hafner, Lt. Comdr. C.T. Sutherlin, and Capt. B.A. Thompson

NCCS Afloat Program Management Office (PMW162)
Commander, Naval Warfare Systems Command, NC-1
Washington, D.C., 20363-5100
Attn: Lt. Comdr. C.T. Sutherlin PMW162-222, Rm. 8E08.
(202) 692-4547

INTRODUCTION

Given the current state of systems development practices, the contemporary Program Manager has three broad management initiatives which he must achieve if the management of his Program is to be successful. Creating a viable management platform requires: (1) the standardization of devices, (2) the reconciliation of conflicting development policies, and (3) the revitalization of the programmatics of systems development.

The Program Office has approached this requirement by combining, in a single Program Office, the four projects: TFCC, ACS, EWCM, and NIPS and by creating a unique development strategy for the amalgamation of the most valuable operational elements of each. From this nucleus of related projects PMW162 has developed a single cohesive application of Information synthesis and management tools called the Naval Command and Control System - Afloat NCCS(A). This paper presents the Operational Requirements, Architecture, SubSystem Elements, Development Strategy, Operational Scenario and a discussion of some of the newer technological innovations of that system.

OPERATIONAL REQUIREMENTS

The operational requirements of NCCS(A) are for mechanisms that allow control of naval forces in not only a physically diverse environment but also in one wherein the elements of control are spread across three distinctly different media: physical, cognitive, and spectral. The physical medium is the "hardkill", or destructive, component of the combat and is the one medium that has been most frequently

addressed. The cognitive medium is the information, or entropic, component of the combat and has only partially been addressed by land based systems. The spectral medium is the "soft-kill", or plasmatic, component of the combat and has never been addressed as an automated system. Thus, NCCS(A) exists because Asset Management for the Battle Force has become intellectually diversified in that it now extends to the control of electromagnetic and information resources as well as of the combatant weapons.

Battle management also encompasses control of two different types of activities each of which requires a different set of behavioral responses. There are those that occur in the First Domain, the actual combat itself, and which could be classed as "paraspontaneous" that is, occurring under conditions of heightened intuitive awareness. Then there those that occur in the Second Domain, the preparation for the combat, and these activities are accomplished in a more methodical evaluative framework. Decision Support and the analytical manipulation of information are activities that are more readily applicable to the measured framework of the planning processes.

While its data correlation and management features are fully operant in the First Domain, the NCCS(A) is focused primarily on the preparation for the battle and hence is a Second Domain tool. In the Second Domain, the preparatory activities include the development of a Plan of Action, and the supervision and modification of Force disposition. It is the Plan of Action and the disposition of the Force that serve as the basis for improvisation during the actual combat. Thus it is the NCCS(A)

products that lay down the operating baseline upon which victory or defeat will be achieved. Using the new software tools of vulnerability assessment, propagation estimation, counter-surveillance, indications and warning, and battle damage assessment provided by NCCS(A), the accuracy and timeliness of Second Domain planning and supervision (i.e., the control of assets and their disposition) can be improved.

Additionally, increases in the range of weapons has resulted in the expansion of the area within which engagement can occur. The relatively unchanged radius of the organic Force sensor envelope does not extend to the effective range of the newer weapons. This results in an area wherein the newer weapons provide a kill capability while existing Force sensors provide no surveillance mechanisms suitable for targeting. This set of circumstances places an increased emphasis on the integration and use of information from nonorganic sensors and National assets. The correlation of this additional information with that of organic sources has increased the information processing requirements during the Second Domain. It should be noted at this point that the term "Information" (as opposed to "data") implies intelligibility and therefore a certain amount of human interpretation. Accordingly, there is a new dependency on data correlation and information management within the Composite Warfare Commander's (CWC) team. NCCS(A) supplies the tools with which these data will be correlated and interpreted for display as "Information".

Finally, since the Commander's dependence on National intelligence information, platform sensor data, and message communications is heightened, he must also protect the electromagnetic envelope of his forces as thoroughly as he protects the physical integrity of his ships. Complementary features of the NCCS(A) tool also provide him the means for controlling this "Electromagnetic Battle". NCCS(A) is, in reality, a fully integrated EW Decision Support System whose analysis modules are pertinent to spectrum management as well as to information fusion. These interactive processes provide EW vulnerability and countermeasures analysis, readiness assessment and asset control.

Thus, the NCCS(A) product as defined by the current PDW 162 procurement strategy will accomplish all of the information management and decision support functions required by the Electronic Warfare Coordinator. It will accommodate all of the functions of Planning, Assessment and Supervision inherent in the management of the electromagnetic battle during the Second Domain. It will provide the basic algorithmic support for data correlation and interpretation across the organic/non-organic and Genser:SCI components in both Domains. Further, it will decentralize the performance of these tasks through the use of a suite of standardized, integrated Advanced Color Workstations.

Because its MMI is performed on 8086 based NDI equipment (i.e., off-the-shelf items), additional features that are unique to each warfare area specialty and to each

subordinate Staff specialist can be provided through transportable software programs. The product being created under this development initiative, will allow analysis packages that support any of the CWC watchstanders to be loaded and applied at any station within the NCCS(A) equipped spaces. With its integrated, standardized suite of NDI equipment hosting transportable, modular, processing software, the NCCS(A) is truly the CWC's battle management tool of the 90's. It provides a solid basis for the measured introduction of Decision Support and Artificial Intelligence Technology into the tools of tactical direction and battle management that are emerging in the Fleet.

SYSTEM ARCHITECTURE

The functional, logical and physical architecture of the NCCS(A) suite are consistent through all of its development phases. As will be seen in the discussion of the procurement strategy for the prototype, interim, and "full-up" versions of NCCS(A), the following architectural designs form the basis for all aspects of its evolutionary development.

Functional Architecture

Functionally, the NCCS(A) system is a Group Decision Support system that is composed of a number of Advanced Color Workstations connected by a multi-level security, dual ring bus. These workstations, though standardized in respect to I/O, display and Operating system, contain sufficient CPU and memory to act as individual information processing systems. Accordingly, the NCCS(A) functions are distributed throughout a network of interchangeable workstations served by a centralized Relational Database and Communications facility. These latter components are hosted in a 32 bit, parallel processing environment whose principle responsibility is to provide data correlation and track management support.

Figure 1, The EWCM DSS Architecture, presents the functional architecture of a single decision support component of the NCCS(A) as it appears to the User of a standard workstation - e.g., the Electronic Warfare Coordinator (EWC). The EWC has facilities for processing his Electronic Battle Management analysis functions from his Battle Station, or from any of the spaces served by the NCCS(A) LAN (i.e., its dual ring bus). On it he can iterate different solutions to EW problems by invoking differing mathematical algorithms and analysis modules as decision support tools. Other than data retrieval and downloading the Common Tactical Picture, these analyses can be completed without burdening the rest of the NCCS(A) system.

Invoking various algorithms and models in a coherent, integrated manner requires a Model Base Management System of some kind. Addressing the implementation of this function, Sen and Biswas have stated: "... at the strategic planning level there is a great need for different models to analyze and support decisions. These models need to be integrated automatically... Integration at the physical level... requires that the users have a lot of knowledge about implementation details. However,

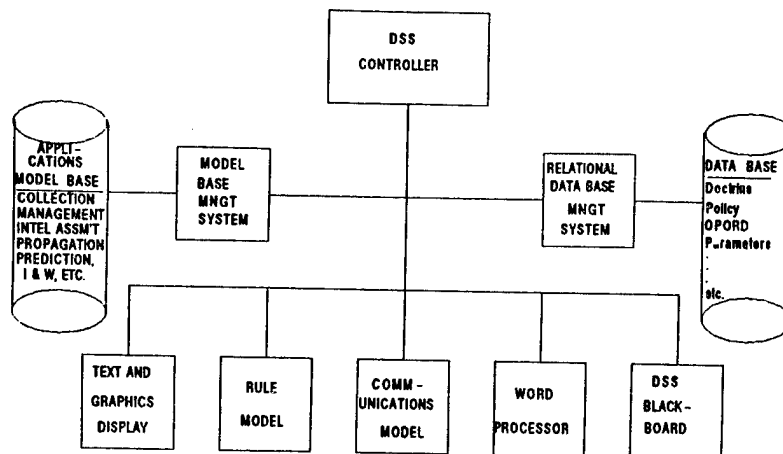


Figure 1
The EWCM DSS Architecture

integration of models at the logical level using knowledge-based systems techniques should make the package independent of the particular system [e.g., workstation] they are implemented on" (1). However, while Sen recommends Expert Systems techniques for concatenating the various analysis models, a simpler approach using linear modeling has been proposed by Pieng-Laing (2). Undoubtedly a synthesis of these approaches will be applied in the mature version of NCCS(A). In the prototype version, at least, various amounts of operator intervention will be required.

Accordingly, the functional architecture depicted in Figure 1 illustrates that by downloading appropriate applications modules into the Model Base Management System of his processor, and through distributed access to his partition in the Relational Database Management System, the EWC can invoke Rules Models and Decision Support "blackboard" functions that are unique to his analysis responsibilities. Standard text and graphics display, word processing, and intra-/extra-NCCS(A) Communications modules complete the functional architecture of the workstation. Complex processes such as track management, data correlation and information fusion that are common and/or too demanding for the workstations are processed centrally and the pertinent results provided to Users via the centralized Database Manager.

Logical Architecture

The logical architecture of the NCCS(A) is illustrated by its treatment of battle management information as it is created from the correlation of data from organic and non-organic sources. Figure 2, The EWC's Information Preparation: Organic & Non-Organic Fusion, presents five levels of information fusion that culminate in its incorporation in the Planning, Assessment and Supervision functions inherent in Second Domain Battle Force control:

Levels one and two portray the collection of data and its correlation as

similar source tracks that are disseminated on standard NTDS communications media. Level 3 represents the fusion of these data in the NCCS(A) correlation function where dissimilar source tracks and parametric data are fused. At level four, the multi-level security feature of the dual ring bus (i.e., LAN) facilitates the further integration of information across security compartments within the NCCS(A) suite. Here GENSER and SCI tracks are fused with polished information from national sources.

The final stage of NCCS(A) information management is the application of the fused information product in the Battle Management process as its Common Tactical Picture. This stage is its use by various members of the CWC Team (in this illustration the EWC), operating at their respective workstations, as an adjunct to their own analysis algorithms and task specific data facilities described above.

Physical Architecture

The physical connectivity of the system is illustrated in Figure 3, the: "CV NCCS Afloat Network". The NCCS(A) strategy for providing Group Decision Support to the CWC team is accomplished by distributing its standardized workstations throughout CVIC, SUPPLOT, TFCC, and CIC. The "Mainframe" processors are centralized in the CVIC space and, as has been noted, provide centralized database management, communications, data correlation, and information integration support. The Large Screen Displays in the three principle CWC work areas facilitate briefing, coordinated analysis, and the formation of group consensus. Consensus formation is enhanced because the standardized display function of the workstations includes a "windows" capability. When this ability to split the screen for the display of results from several tasks is invoked on the Large Screen Display it will enable the simultaneous presentation of planning considerations from several (overlapping) warfare areas. This capability is one manifestation of the NDI management tools that will be provided by

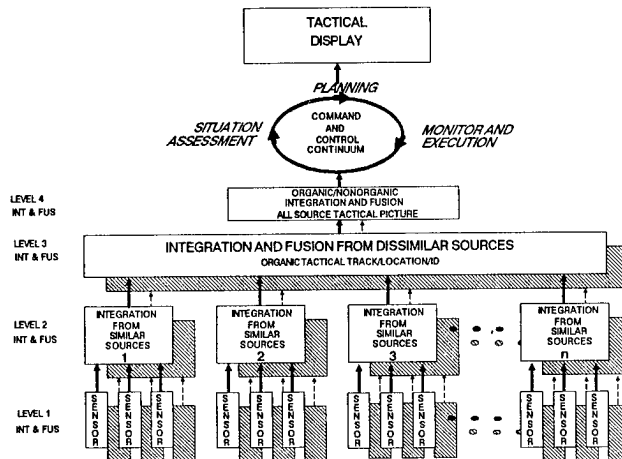


Figure 2

The EWC's Information Preparation: Organic & Non-Organic Fusion

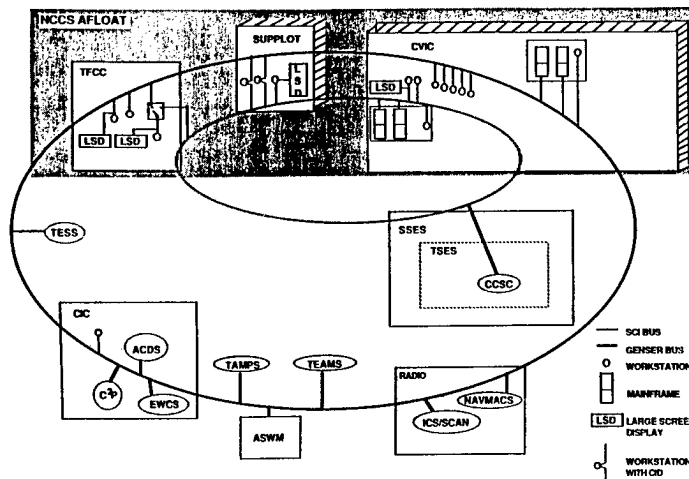


Figure 3

CV NCCS Afloat Network

the Group Decision Support component of NCCS(A).

The distributed components of the system are connected by a dual ring, multi-security-level bus that operates under a Token control protocol. Two fibre-optic transmission paths connect all components of like security. Each device is connected to the dual bus through a LAN Interface Unit (LIU) which includes a bypass device for each ring. Each bypass has the capability to act as a switching node and thus each device is logically connected to both rings. The bypass capability also allows multiple redundant paths of transmission in the event of device failure or cable interruption. Accordingly, in the event of the simultaneous interruption of both cables at the same node (i.e., the destruction of a compartment) the remaining nodes are automatically reconstituted into a single ring.

Viewed from the perspective of security-levels, a set of four fibre-optic transmission paths exist wherein two GENSER rings and two SCI rings provide physically separate, mechanically redundant data exchange paths. Cryptologic interface devices couple the dissimilar security links and thereby provide a certified, sanitized path of information between SCI sources and users in GENSER spaces. External communications are supported by connections to either of the segregated rings and a Battle Override is provided whereby one workstation and Large Screen Display in TFCC can be manually connected to the SCI bus.

COMPONENT MODULES

Addressing the tools of decision-making, VAdm. J. Boyes (3) once noted: "... two human levels of [Navy] decision-making must be separated ... the commander

... [and] staff support to senior operational decision makers." The Navy's emerging NCCS(A) support for the Composite Warfare Concept of battle force management recognizes that concept - particularly so in the case of EW data and Information fusion. In the management of EW data, as it flows from the collector (sensor) to the senior commander (User), raw data is transformed into information in several stages of command activity.

The Composite Warfare Commander, as flag officer in tactical command is supported by other specialized commanders. The air, surface, subsurface and strike warfare commanders command "by negation" the various elements of combat within their purview. The CWC reviews the overall evolution of the battle through the medium of his specialized staff. The EW staff officer, the Electronic Warfare Coordinator (EWC), has responsibility for all EW and C3CM activities across all of the warfare areas. He is responsible to them and to the CWC for all EW and C3CM plans and doctrine in support of the mission.

Active and passive sensors aboard the various force platforms collect data on adversary activities. These data represent statistically independent events and are analyzed and interpreted at the mid-management level by each platform's EW Supervisor. These synthesized elements of "meta-information" are then incorporated into the NTDS(ACDS) component of the TSS. Through this medium they become available to the NCCS(A) information management suite where they are integrated and fused with intelligence data from national sources.

Using the resulting high confidence information, staff officers assess Force plans and activities and recommend appropriate modifications to the CWC. This section discusses the major units of information analysis in the NCCS(A) suite. These are the Afloat Correlation sub-System (ACS), the Electronic Coordinator's Module sub-System (EWCM) and the decision support components of each. EWCM, being created within NCCS(A), is the first of a family of decision support tools that are being developed for use by the CWC staff analyst/specialists. All of these decision tools will overlay the Common Tactical Picture presented to the Battle Force Commander by its ACS component.

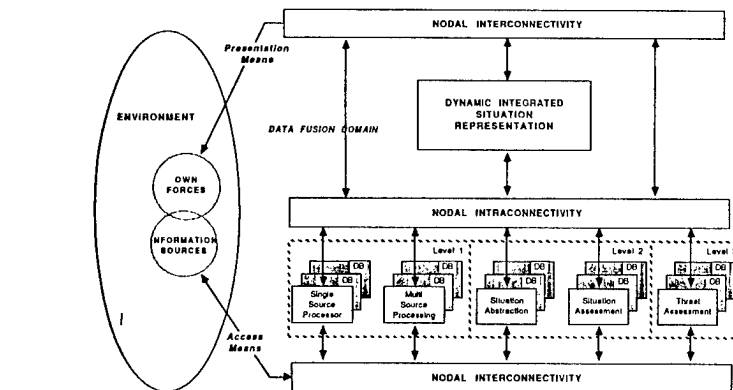
Electronic Warfare Coordinator's Module (EWCM)

The EWCM subsystem will provide automated plans development, monitoring and supervision to the EWC. It will also allow continual re-evaluation of plan suitability and assessment of alternative actions. When the EWCM sub-system is fully integrated into the NCCS(A) information management suite, it will accept interpreted parametric data from platform sensors and combat direction systems. It will receive force status reports, equipment availability and transmission schedules, intelligence estimates and atmospheric measurements. The EWCM partition of the database will house Force Operations Orders, Rules of Engagement, Order(s) of Battle and intelligence data.

Using EWCM, the EWC can evaluate the composite meaning of the various parameters of organic and non-organic electromagnetic data. He can estimate the effects of existing atmospheric and environmental conditions on the propagation characteristics of both his own, and his adversary's EW and C3CM devices. By analysis, the EWC will be able to investigate the probable effects of committing some of his weaponry without the need to retain a great deal of simulated data in his own mind. He will not, of course, burden the C2 systems with these trial solutions. At the same time, he will be able to visually observe the impact of his prospective decisions as overlays on the Common Tactical Picture and as quantitative Utility values on the Decision Blackboard.

Afloat Correlation System (ACS)

The centralized, parallel processing components of NCCS(A) which deal with data correlation and information fusion comprise the ACS subsystem. The workstation analysis modules can call upon these components for the results of Organic and Non-Organic Track Correlation, and for Collection Management, Indications and Warning, and Targeting Support. Figure 4: "Data Fusion Domain", prepared by the JDL Data Fusion Subpanel, presents a taxonomy of fusion activities as they relate to the logical functions of Track Management.



*FROM: DATA FUSION TAXONOMY
PRODUCED BY JDL DATA FUSION SUBPANEL

Figure 4

Data Fusion Domain

The ACS will approximate these functions at levels 1 and 3. The track correlation function of ACS is centralized in its parallel processing mainframe. Threat assessment, intelligence assessment support, collection management, and indications and warning generation are decentralized functions that are distributed across the NCCS(A) suite. Table 1, The Functional Distribution of the ACS Sub-System Activities, presents the allocation of ACS tasks between the centralized processing and the workstation facilities.

When considering the centralization-decentralization of NCCS(A) elements, it will be remembered that there are several aspects of these elements which can be controlled or distributed. Beside the relative distribution of actual computations (i.e., processes) the standardization of workstations provides the potential for distributing and thus decentralizing logical (i.e., operational) functions, intellectual (i.e., analytical) activities, and authentication (i.e., certifying) responsibilities. Thus, the ACS sub-system, as incorporated in the integrated NCCS(A) design will centralize processor intensive data function and track correlation processes and distribute the management activities of situation assessment, planning, and control among the contributing workstations of the CWC staff team.

DECISION SUPPORT MECHANISMS

One of the most forward looking features of the NCCS(A) system is the incorporation of its decision support components. A Decision Support System (DSS) is a man-machine couple that facilitates the incorporation of experience and instinct

in decision-making. It enhances the process of choosing by using data recall, sophisticated data manipulation and graphic display to stimulate managerial evaluation of a set of choices. Decision Support systems are unique among computer based information systems in their ability to apply ad-hoc simulation as a medium for hypothication (What If?) and automated goal seeking to the solution of problems.

The dichotomy of combat decision-making identified by VAdm. Boyes is well served by providing these capabilities at the CWC level. Table 2, The Characteristics of Hierarchical Decision Making, is based on Simon's pioneering work in organizational behavior (4). It compares the levels of management in a Battle Force with the intentions (Activity Orientation), data, and decision processes they entail and with the automated tools which support them. It illustrates that at the Flag level, as the decision process becomes less structured, the criteria upon which the choices are based are best served by the Decision Support System technology that is emerging in contemporary Information Systems applications.

The definition of structured problems provided by Jones offers an interesting counterpoise with which to delimit unstructured decisions: "A structured problem is one ...[for which] decision rules can be specified that allow for problem definition, designation of alternatives, and selection of the best alternative" (5). Clearly, this exposition illustrates the dichotomy of the NCCS(A) problem solving dilemma. On the one hand, the system itself is a decentralized group consensus forming tool serving the CWC team. On the other, it is both a calculating tool and a personal decision support system. It not only calculates

Table 1
The Functional Distribution of the ACS Sub-System Activities

Functional Category	Specific Functions	Distribution
Sensor I/O	<ul style="list-style-type: none"> * Accept Sensor reports * Accept data linked remote sensor data * Provide common time/nav reference * Monitor status of sensors 	Centralized
Sensor Control	<ul style="list-style-type: none"> * Coordinate multiple sensor searches * Cue and handoff sensor-sensor data * Coordinate emitter muting * Prioritize sensor activities * schedule sensor events, modes * Provide threat alerts: hostile contacts * Report data base (data, history, quality) * Accept sensor cues 	Centralized
Data Fusion	<ul style="list-style-type: none"> * Associate data from all sources * Maintain temporal target tracks * Combine associated feature data to: <ul style="list-style-type: none"> * Improve location accuracy * Remove identification ambiguity * Increase confidence in data * Estimate temporal activity of periodic emitters or emitter complexes * Emitter kinematic tracks * Emitter activity history 	Centralized
Situation Assessment	<ul style="list-style-type: none"> * Monitor all-source contact data base * Detections, tracks, files, events * History file * Aggregate multiple targets into threat complexes * Develop order of battle assessment * Prioritize threats and opportunities * Monitor ownship emissions for emcon compliance * Maintain platform/attribute data base * Provide survivability assessment 	Decentralized

Table 2

The Characteristics of Hierarchical Decision Making

MANAGEMENT ACTIVITY	ACTIVITY ORIENTATION	DATA			DECISION			DECISION SUPPORT TOOL
		SOURCES	FORMAT	CURRENCY	SITUATION	PROCESS	CRITERIA	
FLAG PLANNING TOP MANAGEMENT	ACCOMPLISH HQ MISSION	EXTERNAL	OVERVIEW	HISTORICAL	UNIQUE	NON- STRUCTURED	INTUITIVE	DSS
PLATFORM PLANNING AND MANAGEMENT	ALLOCATE AND EMPLOY RESOURCES	MIXED SOURCES	PLATFORM SPECIFIC	MIXED				C3 SYSTEM
WEAPONS AND SENSORS OF CONTROL	EXECUTE MISSION OBJECTIVES	INTERNAL SOURCES	SYSTEM SPECIFIC	REAL TIME	RECURRING	STRUCTURED	OPTIMIZE	WEAPONS SYSTEM

track and data correlation probabilities but it also allows the individual decision maker access to algorithms for solving the structured components of his warfare problems. Then, the individual structured problems having been solved, in the aggregate (through the Blackboard technique) it becomes a personal decision support system at the workstation level. Finally, combining the CWC Staff activities on the individual workstations into the comprehensive NCCS(A) system, the group consensus itself becomes the decision support tool that serves the CWC in the solution of his own unstructured problem(s).

Decision Modules

The EWCM sub-system component of NCCS(A) will include components that house the following existing decision modules: Vulnerability Assessment Device (VAD), Integrated Refractive Effects Prediction System (IREPS), the HF prediction system - PROPHET, Communication Plan Automation System (CPAS), and a radio frequency de-confliction algorithm. Ideally, the full-up version of NCCS(A) will be constructed to allow these, and any algorithms that are created in the future, to be embedded without modification. It is unlikely that the interim EWCM will reach this level of sophistication but the concept of insertable applications modules is certainly a candidate for P3I and is being considered in the ongoing detailed definition.

Model Base Control Functions

The EWCM is designed to allow the operator to make iterative judgements and incremental evaluations of the effects of decisions. Accordingly, a key feature of the model base management system is its capability to format and transpose data elements from one application module for use within another. This concept not only provides the logical integration that will unburden the User but it will also ensure the capability of EWCM to expand its model base as increasingly sophisticated analysis algorithms become available. It guarantees that the completed NCCS(A) system will be a receptive host to the analysis packages that will ultimately be developed by the other CWC staff specialties.

Temporary recall and replay of previously considered simulation scenarios will also be required of the model control function. A scratchpad will be allocated for the storage of several iterations of an ongoing analysis. Parametric changes will be logged and available with synthesized alternative plan elements temporarily saved. This feature will enable the User to review the process by which he has arrived at his latest proposed solution to every unstructured decision.

The Blackboard Device

The present NCCS(A) application of DSS technology is primarily concerned with the active and passive use of the electromagnetic spectrum through the creation (and modification) of spectrum use plans and frequency allocations. Notwithstanding the rigors of its creation during the early part of the Second Domain, the process of maintaining changes to the plan for frequency use/surveillance is a process that entails many sequential and inter-related decisions. Without a comprehensive blackboard tool, under conditions of imminent combat when the electronic battle is at its most intense, the correlation and assimilation of so large a number of dis-associated decisions may limit the utility of the system. Accordingly, a simplified presentation medium must be developed. One model that has been proposed is an adaptation of the techniques of Expected Value theory. This paradigm offers both a memory aid and a decision correlation tool. This concept, developed as a decision tree and applied to the concatenated decisions of spectrum planning is illustrated in Figure 5, The Blackboard: Planning Tree.

This representation presents the decision-maker with an initial decision recommendation along with the audit trail of all its precursor decisions. With the decision tree solution displayed on the screen, the DSS user can evaluate the proposed solution according to his perspective of the combat situation as a specialist-member of the CWC team. Following his review of the proposed optimal solution the user can then evaluate all of the intermediate decisions (i.e., nodes of the tree) that contributed to its selection. In so doing he can accept

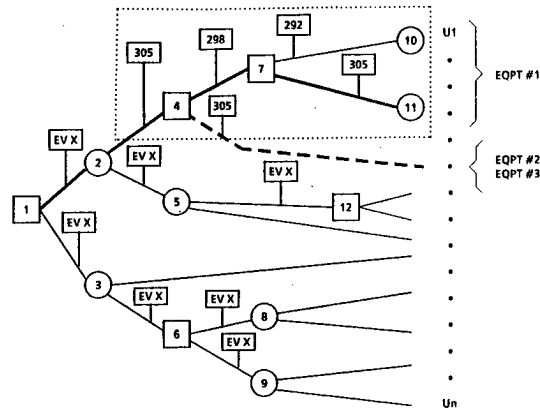


Figure 5

The Blackboard Planning Tree

each machine generated solution or he can modify any of the elements of that solution. For example, the decision matrix of a frequency assignment paradigm could be reviewed and a subjective weighting of payoffs assigned according to the operator's judgement. These weights can be combined with the Expected Values of the decision node. This process, in effect, transmutes the mathematically structured stochastic approach of Decision Theory into a solution that encompasses the behaviorally cognizant elements of Prospect Theory (6). Alternatively, the user's review of the prediction mechanisms might call for an application of Bayes theorem and he might momentarily invoke a model of the theorem to aid in his adjustment of the probability estimate.

The vehicle for this interactive modification is the decision tree portrayal of the problem set. Its manipulation provides a "blackboard" that incorporates the instinctual and intuitional processes which are the basis of Decision Support technology. There has apparently been some success in applying a variation of this technique to the Group Decision process. Johnston reports: "The Group Decision Aid guides the decision making process through the selective evolution of a multi-level decision tree, consisting of possible actions and events resulting from them ... the system asks the individual group members to input their estimates of the likelihood as well as the value of specific decision outcomes" (7).

APPLICATION

Decision Support is first of all, a man-machine interaction. As has been said, it is a process of data manipulation which is assisted by automated tools and which encourages the User to reach a conclusion about his unstructured problem solution. Bennett succinctly puts it: "A recurrent theme in DSS (Decision Support System) research is User learning.... A DSS does not solve problems; it lets individuals exploit their own skills in problem solving. The obvious strategy for DSS design is to support first and extend later" (8). Recursive development of a problem solution is implicit in the Decision Theory underpinnings of the Blackboard mechanism.

The following example is an illustration of a hypothetical man/machine interaction in electronic battle management. It suggests both the approximate interactive decision play between user and machine and the implementation of the Decision Theory concept underlying the Blackboard device discussed above. The scenario is presented in terms of a matrix solution to the series of successive decision problems. These problems could be concatenated however and the composite presented as shown in Figure 5. A "SelectZoom" effect (as illustrated) could be used by the End-User to decompose the problem into its component elements for selective treatment of its various technical and/or operational aspects. This application presents the standard one shot Decision Theory technique to the assignment of frequencies. The scenario extends over three time periods for the purposes of illustrating the impact of additional data upon the solution:

"It has become necessary to create a net for communications with several distant platforms that are moving according to a predefined schedule. The Electronic Warfare Officer has exercised the vulnerability prediction algorithms of his DSS and has determined that there are two modes of communication which will fulfill his requirement within an acceptable level of ECM/ECCM risk. He has the option of establishing a HF or of using a special national channel. It is likely that authorization for the use of the national asset will not be forthcoming so will he first evaluate the HF net requirements."

Among the interrelated elements involved in the selection of an appropriate HF transmission frequency are: frequency availability, equipment capability, broadcast range, and power levels. Compounding the difficulty of the choice are various transmission variables that change according to location and time of day. In this illustration of DSS use of a blackboard, the user intends to select from among three authorized frequencies the one that best suits the needs of the force net. He has algorithms available which provide range characteristics from a propagation prediction model, environmental parameters from an automated atmospheric sampling

system, and an algorithm which addresses the problem of harmonic interference.

Case 1. A set of three frequencies is available for the assignment to the force HF network. The net must achieve a transmission to a series of moving platforms. It has been determined that the solution payoff must attain a figure of merit of at least 300 for reliable transmission to occur. Combinations of the predictions of a series of embedded stochastic tables such as: Surface Duct phenomena, Enhanced Range predictors, Elevated Duct phenomena, sunspot activities, and the output from a transmission path prediction model provides the following estimates of the occurrence of a skip effect:

S1 < 300 nautical miles; p=.35
S2 > 300 but < 1000 nautical miles, p=.4
S3 > 1000 nautical miles; p=.25

Expected Distance of Transmission Skip
Versus Frequency Assignment (case 1)

State of nature (p)	S1	S2	S3	Expected Value (EV)
Frequency Assigned	.35	.4	.25	
f1	230	185	365	246
f2	375	320	100	284
f3	150	100	800	292 <-

Note: Payoff figures of merit are assigned for illustration and should be developed through separate research (see: Composite EW Quantifiers, Ormsby (9))

The Expected Value (EV) column presents the weighted figure of merit for each frequency. Note that none of the expected payoffs meets the required minimum figure of merit of 300. Using only this "screen" the operator would select frequency f3 as being the best choice even though the required parameters of communication were not met. The principle of choice for differentiation between f3 or f2 is not so clearly evident however. Trade-offs between these two frequencies that are not contained in the present problem are not obvious and thus the operator is prevented from considering them in this decision. Accordingly, in the absence of any other information, the operator is reduced to making the un-enlightened selection of f3.

Case 2. In addition to choosing a frequency, the operator must select from among three power settings in consonance with equipment performance profiles, EMCON restrictions, and ECM intercept predictions. The states of nature and Figures of merit are derived from reliability estimates, duty cycles, atmospheric predictions.

Expected Performance Profiles at
selected power Outputs (case 2)

State of nature (p)	S1	S2	S3	Expected Value (EV)
Power Output	.60	.15	.35	
po1	230	185	365	257
po2	375	320	100	298
po3	150	100	800	305 <-

Note: Payoff figures of merit are assigned for illustration and should be developed through separate research (see: Composite EW Quantifiers, Ormsby (9))

The appropriate choice here is clearly po3. However, (as an elaboration of the decision support concept) the operator may have reservations about the interrelation between the selection of the frequency and the power output as they relate to the potential contribution the effect of the skip made to the Expected Value of the figure of merit calculations. This could result for instance, from his past experience in a similar situation when the predicted phenomenon failed to occur. In this case he might choose ps2 since it almost met the minimal selection criteria and it also (hypothetically) satisfies his own instinctive evaluation of the situation. The principle of choice here is "Satisficing" since the selected frequency does not meet the minimum criteria but the match is close enough to preclude the expenditure of more resources in creating another solution.

Case 3. In the final iteration of the example, the operator consults the "Operational Events" and movement reports to determine the anticipated positions of the platform he desires to have communications with. He observes that the formations planned for the Force at this timeline will probably also invalidate the use of f3 which depends very heavily on the effects of the skip. While he contemplates this line of reasoning, he consults equipment availability files to assess the various equipment and emitter combinations (EE) available. He then causes the harmonic effects prediction algorithm to evaluate three of the combinations in relation to the probability of interference from platform reflected harmonics (H) at f3, po2. Predictions of intermodulation are particularly stringent in the case of f2 and, as a result of this analysis (hypothetically) effectively reduces its radiated power. The harmonic effects predictor creates the figure of merit payoffs shown below.

Expected Harmonic Interference Versus
Equipment-Emitter Availability (case 3)

State of nature (p)	H1	H2	H3	Expected Value (EV)
Equipment-Emitter	.25	.60	.35	
EE1	230	185	365	296
EE2	130	180	350	263
EE3	150	100	800	305 <-

Note: Payoff figures of merit are assigned for illustration and should be developed through separate research (see: Composite EW Quantifiers, Ormsby (9))

The user is prompted by this analysis to choose equipment-emitter combination EE3. At this point the operator may care to revise his prediction about the advisability of using f3, po2 because the combination of the transmitter equipment and emitter device selected may be unique and therefore not covered by his previous experience.

In reality, harmonic interference would probably effect all of the frequencies and power setting possibilities and would call for a total re-evaluation (a recursion on the present example). In all probability, additional factors would have to be considered as adjustments in power output would require reference to the Rules of Engagement, and EMCON Rules; re-selection of the radiating surface would incorporate EOOB, Table of Equipment, and Equipment availability review; and frequency selection criteria might require a further analysis of platform location(s), the acceptable level of through-put reliability, or the minimum distance criteria.

DEVELOPMENT STRATEGY

NCCS(A) is an evolutionary progression of Battle Force Management tools that are based in the already operational Flag Data Display System (FDDS) component of the Tactical Flag Command Center (TFCC). A modular upgrade of the data transmission facilities of the FDDS and the inclusion of a network of advanced color workstations will, in the near term, result in an early operational capability for dissimilar source data integration and for planning and supervision. As a precursor to this Interim NCCS(A), an extensive validation of the concept is being conducted in an operational environment aboard the USS Vinson.

Various Fleet End-User sources have developed elementary prototype devices that

model some of the elements of the NCCS(A) system. By interconnecting a suite of these prototypes through an Ethernet bus a simulation of the interrelationship of the distributed functions of these components can be established. While it is acknowledged that the aggregate effect of these devices is relatively unsophisticated in comparison with that of the Interim version (i.e., the FDDS upgrade), the End-User reception and use of these devices is being measured as an indication of the applicability of functions of the more sophisticated Interim system.

The final phase of the NCCS(A) development, its "full-up" version will complete the TFCC upgrade cycle. It will incorporate the full set of advanced propagation assessment algorithms hosted in the model base management system. The latter, probably served by a graph based integration mechanism, will enhance the construction of ad-hoc models of spectral behavior. This assessment feature will allow pertinent decision parameters to be individually analyzed and incorporated into an event sensitive battle management solution. Using the Decision Theory concepts found acceptable during the precursor developments, state-of-the-art consensus forming protocols will be embedded in this CWC Decision Support System that will allow group collaboration in the comprehensive management of the Force Energy Spectrum across all of the composite warfare domains.

REFERENCES

1. Sen, A.; Biswas, G.; "Decision Support Systems: An Expert System Approach"; Decision Support Systems; I, pp. 197-204; North-Holland, 1985.
2. Ting Peng-Liang; "A Graph Based Approach to Model Management"; pp. 136-151; Proceedings, International Conference on Information Systems; San Diego, Ca.; Dec. 1986.
3. Boyes, J.; "Intelligent C3I Systems"; pg. 15; SIGNAL; AFCEA Journal; June 1986.
4. Simon, H.; The New Science of Decision Management; Prentice Hall; Englewood Cliffs, N.J.; 1977.
5. Jones, B., Eller, B.; "The Evolution of Decision Support Systems"; pg. 4, Journal of Computer Information Systems; Fall, 1986.
6. Khaneman, D., Tverski, A.; "Prospect Theory: An Analysis of Decisions Under Risk"; pp. 263-291; Econometrica, Vol 47, #2; March 1979.
7. Johnston, S.; "An Interactive Computer Aided System for Group Decision Making"; Proceedings, MIT/ONR Workshop on Distributed Information and Decision Systems Motivated by C3 Problems; IT, Cambridge, Ma.; Sept 1980.
8. Bennett, John I.; Building Decision Support Systems; Pg. 152, Addison-Wesley; Menlo Park, Ca. 1983.
9. Ormsby, J.; "Composite Quantifiers and Methods in Electronic Warfare"; Proceedings, Joint Electronic Warfare Conference; San Antonio, Tx. 1985.

ABOUT THE AUTHORS

Dr. Arnold N. Hafner is a Staff Scientist with a major systems development firm and an Associate Professor of MIS at United States International University. He has published extensively on Decision Technology, Information Management and the Management of Systems Development. Captain B.A. Thompson is a graduate of the Naval Academy, Class of 1964 and has a Master of Science in Electrical Engineering from the Naval Postgraduate School. He is a designated Material Professional and is the SPAWAR Program Manager for the Naval Command and Control System - Afloat (NCCS-A) Program. Lt. Comdr. C.T. Sutherlin holds a Masters Degree in EW Systems Engineering from the Naval Postgraduate School and is serving with the NCCS(A) Program Office. He has done research in the development of Navy EW Systems and is published on the subject of Decision Support in the Electronic Warfare arena.

BELIEF REPRESENTATION FOR FUSION USING EVIDENCE THEORY

John W. Betz

The Analytic Sciences Corporation
55 Walkers Brook Drive
Reading, MA 01867

Evidence theory is especially appropriate for the fusion of data from multiple sources because of its ability to accommodate uncertainty and conflict. A limitation of this powerful formalism in fusion, however, has been the inability to rigorously assign belief measures to assertions based on numerical outputs from sensors. A rigorous approach to representing belief in numerical quantities is presented that extends the applicability of evidence theory in data fusion.

This approach applies to assertions that a numerical quantity takes on values in an interval or on a half-line, extending Dempster-Shafer theory to symbolic information derived from sensor data. The method assumes that incomplete statistical models can be developed for the numerical quantity in the form of a closed set of probability density functions, even when a single density function cannot be assigned because the underlying statistics of the data are not known perfectly and the effects of both the sensor and the processing cannot be modeled perfectly. The representation explicitly accounts for simplifications in analytical models, and for unknown parameters that affect the statistical description.

The derivation of Shafer belief functions from the set of probability density functions is described based on the use of mass density functions, introduced by Strat to represent the assignment of mass to assertions based on numerical quantities. The contribution of this paper is the development of a rigorous, statistically-based, procedure for deriving the mass density function. The necessary and sufficient conditions for the set of probability density functions to produce a Shafer belief function are presented, and it is shown that the lower distribution function originally described by Dempster both is a lower bound on the belief function and has the properties of a belief function. A simple expression for the plausibility function in terms of the upper and lower distribution functions is also derived.

The exposition provides additional conceptual and mathematical linkage between evidence theory and Bayesian theory. In particular, when the set of density functions has unity cardinality, the resulting mass density function is equivalent to a probability density function, and the belief function is Bayesian. Consequently, the variation among members of the set of density functions can be viewed as an expression of ignorance in evidence theory. The analysis thus reinforces the view of Bayesian statistics as a limiting case of evidence theory.

BELIEF REPRESENTATION FOR FUSION USING EVIDENCE THEORY

John W. Betz

The Analytic Sciences Corporation
55 Walkers Brook Drive
Reading, MA 01867

INTRODUCTION

Evidence Theory, also known as Dempster-Shafer Theory, is especially appropriate for the fusion of data from multiple sources because of its ability to accommodate uncertainty and conflict. The belief measures that represent uncertainty in Evidence Theory are both less constraining and richer than probabilistic representations, since uncertainty resulting from ignorance and statistical variability can be separately represented numerically, without the limitations of a single point probability representation. Furthermore, mass may be assigned not only to the atomic propositions that are the mutually exclusive, exhaustive propositions in the frame of discernment, but also to sets of atomic propositions. The ability to assign mass directly to sets of propositions facilitates the fusion of data from different sensors that provide qualitatively different information at different levels. Dempster's Rule provides a consistent mechanism for combining evidence from multiple independent sources, even with vastly different qualities of evidence and with partial conflict. Under certain common assumptions, Evidence Theory may be considered as a generalization of Bayesian statistics, so that when Bayesian representation of uncertainty can be provided, Evidence Theory provides Bayesian results.

The restriction of Shafer's development of Evidence Theory to discrete propositions limits the ability to represent belief in numerical evidence for applications involving the use of sensor data, such as data fusion. Not only has a rigorous, automated mechanism been unavailable for deriving or representing the belief in numerical quantities or in propositions derived from those

quantities, but a representation is lacking that constitutes a natural framework for human experts to express their heuristic belief. A rigorous approach to representing belief in numerical quantities is presented to address these deficiencies. This approach has many desirable attributes, including the possibility of automation, providing a natural representation for heuristic belief, and contributing to the link between Evidence Theory and statistics.

The second section of this paper reviews relevant aspects of Evidence Theory, including upper and lower probabilities, properties of support and plausibility functions, and mass density functions for representing belief in numerical quantities. New results are contained in the third section, which shows how statistical descriptions of numerical quantities may, under certain constraints, be represented as Evidence Theory belief measures. The fourth and final section summarizes the results of this paper.

THEORETICAL FOUNDATIONS

Evidence Theory is based on statistical developments by Dempster, extended by Shafer to form a non-Bayesian theory that distinguishes between objective and subjective information in considering chance events. More recently, Strat developed the concept of a mass density function to represent belief in some types of numerical information. The results in the third section are based on these theoretical foundations.

Upper and Lower Probabilities

Upper and lower probabilities are introduced by Dempster in Ref. 1 to describe the effect of a multivalued mapping between random quantities. This concept, its extension

to conditional probabilities, and its application to the combination of independent sources of information, form the basis for Shafer's work and the resulting mathematical theory of evidence.

The concept of upper and lower probabilities is based on a multivalued mapping, Γ , between a pair of spaces X and W , so that Γ assigns a subset $\Gamma x \subset W$ to every x in X . The problem is to characterize the probability associated with w , a subset of W , in terms of a probability measure, μ , defined on X . When Γ is single-valued, the probability associated with w can usually be expressed directly in terms of μ . When Γ is a set-valued function, however, only upper and lower probabilities can be defined over subsets of W .

For any $w \subset W$, define w^* as the subset of X containing all values of x that are mapped into any set in W that intersects w , that is,

$$w^* = \{x \in X, \Gamma x \cap w \neq \emptyset\} \quad (1)$$

where \emptyset is the null set. Similarly, define w_* as the subset of X containing all values of x that are mapped into any non-null subset of w , that is,

$$w_* = \{x \in X, \Gamma x \subset w, \Gamma x \neq \emptyset\} \quad (2)$$

Consequently, w^* equals W_* , and is the domain of Γ . For any set w for which the probability measure μ is defined on w^* and w_* , the upper probability of w , $p^*(w)$, is defined as

$$p^*(w) = \mu(w^*) / \mu(W^*) \quad (3)$$

and the lower probability of w , $p_*(w)$, is defined as

$$p_*(w) = \mu(w_*) / \mu(W_*) \quad (4)$$

When x is a random variable and W is a subset of the set of real numbers, upper and lower distribution functions, $F^*(v)$ and $F_*(v)$, can be defined for w by

$$F^*(v) = p^*(w \leq v) \quad (5)$$

$$F_*(v) = p_*(w \leq v) \quad (6)$$

where $\{w \leq v\}$ denotes the event $\{w: w \leq v, \forall w \in w\}$. Neither the lower distribution function nor the upper distribution function, however, bounds the probability of the random variable, v , in a finite interval. For a and b finite with $a < b$, $p(a < v \leq b)$ may be smaller than $F_*(b) - F_*(a)$ and larger than $F^*(b) - F^*(a)$.

Evidence Theory

In contrast to the statistical orientation of Dempster's work, Shafer presents in Ref. 2 an integrated theory of evidence based on a philosophical foundation that differs from Bayesian probability. This original exposition is supplemented by sev-

eral more recent descriptions, defenses, and expansions of Evidence Theory in Ref. 3 and Ref. 4. There are also many useful summaries in the literature, for example, Ref. 5 and Ref. 6, augmented by attempts to view Evidence Theory from a statistical viewpoint, as in Ref. 7.

Shafer's developments provide a set of semantics, functions, and operations in a consistent mathematical framework that is structured to avoid the inability of Bayesian measures to distinguish between objective variability, or chance, and subjective variability, or ignorance. An interval representation of belief in a proposition is employed using an ordered pair of nonnegative numbers, each no greater than unity. The ensuing discussion denotes these numbers the support and the plausibility, deviating slightly from Shafer's semantics. The support function indicates how much the available evidence supports the proposition to the exclusion of any other proposition, while the plausibility indicates how much the evidence does not refute the proposition. Denoting the support for proposition A by $S(A)$, the plausibility by $P(A)$, and the complement of A by \bar{A} ,

$$P(A) = 1 - S(\bar{A}) \quad (7)$$

and

$$S(A) \leq P(A) \quad (8)$$

When $S(A)$ is equal to $P(A)$, the belief is Bayesian and has all the standard properties of probabilities. The difference between plausibility and support is termed the ignorance, and indicates the inability to commit to either the proposition or its complement.

All propositions of interest are subsets of the frame of discernment, which is equivalent to a probabilistic sample space. In contrast to probability theory, mass representing belief can be assigned to any element of the power set of the frame of discernment, not just the exhaustive, mutually exclusive propositions. If a unit of mass is distributed among the propositions, the support for a proposition consists of the mass attributed to that proposition and to its subsets, while the plausibility is the support plus all the mass attributed to supersets of the proposition.

A support function is defined in Ref. 2 as any function that has the following properties on a frame of discernment θ ,

$$S(\emptyset) = 0 \quad (9)$$

$$S(\theta) = 1 \quad (10)$$

$$S(A_1 \cup A_2 \cup \dots \cup A_n) \geq \sum_{\substack{I \subset \{1, 2, \dots, n\} \\ I \neq \emptyset}} (-1)^{|I|+1} S\left(\bigcap_{i \in I} A_i\right) \quad (11)$$

when the $\{A_i\}$ are subsets of θ , and for every positive integer n , where $|I|$ is the cardinality of I . The property in Eq. 9 is

called superadditivity, and is essential to the development in the following section. A plausibility function is defined by the combination of Eq. 7 and the properties of a support function.

Mass Density Functions

The development of Evidence Theory has emphasized finite frames of discernment. Numerical quantities on the set of real numbers or some infinite subset clearly violate this assumption. Mass density functions, introduced as mass functions in Ref. 8, provide an elegant mechanism for representing and manipulating belief that a continuous quantity takes on a value within specific intervals of the set of real numbers. Two-dimensional mass density functions, the focus of the present paper, represent the mass assigned on a frame of discernment that consists of a half-open interval or half-line on the real number line, or the set of real numbers itself. The set of focal propositions, or propositions to which mass may be assigned, is constrained for two-dimensional mass density functions to those of the form

$$w = \{v \in (a, b]\} \quad (12)$$

Mass assignments to sets of non-contiguous intervals can be represented by mass density functions of high dimension. In many applications involving the use of numerical data, however, interest is confined to propositions involving numerical values occupying contiguous intervals, and a two-dimensional mass density function is adequate.

A two-dimensional mass density function $M(x, y)$ is any nonnegative function with unit volume that is zero on the half-plane $y < x$. This is consistent with the defined relationship between a Shafer support function and a mass density function, where the support function for the proposition in Eq. 12 is formally denoted $S[\{v \in (a, b]\}]$ and abbreviated $S(a, b)$. A mass density function, $M(x, y)$, is defined formally by the integral relationship from Ref. 8,

$$S(a, b) = \int_a^b \int_x^b M(x, y) dy dx = \int_a^b \int_a^y M(x, y) dx dy \quad (13)$$

Eq. 12 combines with Eq. 13 to imply that the mass density function is zero on the half-plane $y < x$. Both the region of support for a mass density function and the region of integration for computing the support function are shown in Fig. 1. Concentrating the mass near the hypotenuse in Fig. 1 assigns mass to narrow intervals, indicating the ability to describe accurately the behavior of the numerical quantity. Mass assigned near the upper left-hand corner indicates inability to describe detailed behavior. From Eq. 7 and Eq. 13, the plausibility function expressed in terms of the mass density function, is

$$P(a, b) = 1 - S(-\infty, a) - S(b, \infty) \quad (14)$$

or

$$P(a, b) = \int_{-\infty}^b \int_{\max(a, x)}^{\infty} M(x, y) dy dx = \int_a^{\infty} \int_{-\infty}^{\min(b, y)} M(x, y) dx dy \quad (15)$$

where the region of integration is also indicated in Fig. 1. The support for the union of the two disjoint intervals is equal to the sum of the supports for each interval, consistent with the constraint of a two-dimensional mass density function. Eq. 13 is used as the basis for describing and deriving mass density functions, to build on Shafer's extensive analysis of the properties of support functions.

When the partial derivatives of the support function with respect to the interval bounds are defined, the mass density function can be expressed directly from Eq. 13,

$$M(x, y) = -\frac{\partial^2}{\partial x \partial y} S(x, y) \quad (16)$$

Because the support function is bounded, all constant terms and terms that are functions only of either x or y in the mass density function are zero.

The definition of mass density functions in Eq. 13, combined with the properties in Eqs. 9, 10, and 11, yields two important properties of mass density functions: normalization and nonnegativity. Normalization states that a mass density function has unit volume,

$$\int_{-\infty}^{\infty} \int_x^{\infty} M(x, y) dy dx = 1 \quad (17)$$

This property follows from Eq. 10 and the fact that the frame of discernment is either

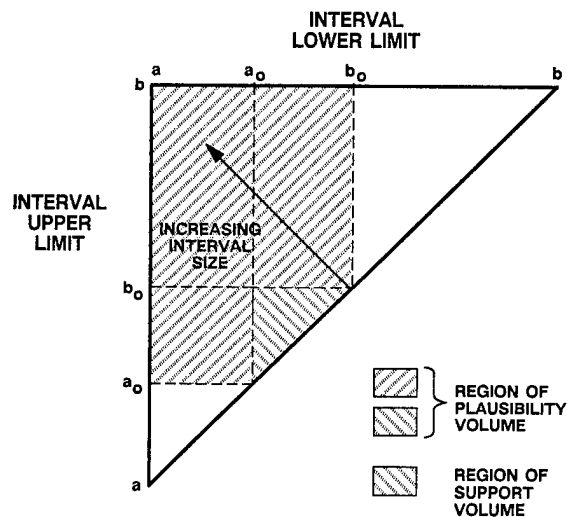


Figure 1 Mass Density Function Regions of Integration

the interval $(-\infty, \infty)$ or some subinterval, so that $S(-\infty, \infty)$ equals unity.

Nonnegativity of the mass density function,

$$M(x, y) \geq 0, \forall x, y \quad (18)$$

follows from the superadditivity of Shafer's support function in Eq. 11. Ref. 9 shows that superadditivity of the support function is equivalent to nonnegativity of the mass density functions.

DERIVATION OF BELIEF IN NUMERICAL QUANTITIES

The elegant representation of belief in continuous quantities by mass density functions tends to be a very unnatural way for algorithms or human experts to directly express belief. While Ref. 8 discusses relationships between possibility functions in fuzzy set theory and the derivation of mass density functions, this approach may be no more natural for representing the belief in sensor data. Sensor performance is typically described statistically, and a mechanism for relating statistical descriptions to mass density functions is expected to be most natural and useful. When there is complete knowledge of the statistical behavior of the numerical quantity, there is no ignorance of the quantity's behavior, and the random behavior can be described by a probability density function. However, when there is a lack of knowledge of the statistical behavior, one way to express the available knowledge is to provide a set of probability density functions whose elements each represent possible random behavior of the quantity. For example, different statistical characterizations may describe sensor data acquired under different conditions. The conditions associated with a particular set of sensor data may not be known, and estimating the information from the data may be impossible or impractical. Under certain conditions, the expression of uncertainty as a set of probability density functions can be used to derive a mass density function.

Fundamental Result

Consider a set V of random variables characterized by different probability density functions, where \mathcal{F} is the set of probability density functions that describe all the random variables in V . Each random variable in V can also be described by a different single-valued mapping from a random variable y that is uniformly distributed on $(0, 1]$, and Γ' is defined to be the set of these mappings. If x is the set of all subintervals, x , of $(0, 1]$, Γ is a mapping of X defined by

$$\Gamma x = \{\Gamma' y, y \in x \subseteq X\} \quad (19)$$

and if W is the set of all subintervals of the range of Γ' , then Γ is a multivalued mapping between X and W as discussed in Ref. 1. The probability measure on X is defined as the probability that y is in x , so the probability of the domain of Γ is unity. The set of probabilities assigned to any w in W , where w is defined in Eq. 12, is composed of the probabilities that v is in $(a, b]$ for every v in V . The probability of w can be bounded by the upper and lower probabilities defined in Ref. 1. Using Eq. 2 and Eq. 4, the lower probability for w equal to $(a, b]$ in W is

$$p_*(w) = p(\{y \in w; w_* = \{x : a < \Gamma x \leq b, x \in X\}\}) \quad (20)$$

where for x equal to any $(a_0, b_0]$ in X , the probability measure is

$$p(y \in x) = b_0 - a_0 \quad (21)$$

Typically, the set \mathcal{F} is composed of a countable number of probability density function parametric forms $\{f_i(v; \underline{a}_i)\}$, each with a possibly uncountable number of parameter variations $\{\underline{a}_i\}$, where each element \underline{a}_i of the vector \underline{a}_i describes a different probability density function parameter, and is bounded by the interval $[a_{ij}^{\min}, a_{ij}^{\max}]$. Under conditions to be defined, the set \mathcal{F} can be used to derive the support and plausibility for an interval w , based on Theorem A.2 in Ref. 10. The theorem states that when a closed convex set of probability functions is defined over the frame of discernment, and for every set of propositions on the frame of discernment, A_1, A_2, \dots, A_n ,

$$\min \left[p \left(\bigcup_{i=1}^n A_i \right) \right] \geq \sum_{I \subseteq \{1, 2, \dots, n\}} (-1)^{|I|+1} \min \left[p \left(\bigcap_{i \in I} A_i \right) \right] \quad (22)$$

when the minimum is taken over the set of probability functions, then for every proposition A in the power set of the frame of discernment,

$$S(A) = \min[p(A)] \quad (23)$$

In the application of this theorem to the representation of belief in continuous quantities, interest is confined to propositions relating to contiguous intervals. Consequently, the number of elements in the frame of discernment, n , in Eq. 22, can be restricted to two. Moreover, the requirement that the set of probability functions be closed and convex is sufficient and not necessary. This theorem holds whenever there exists a unique minimum probability for every set of propositions on the frame of discernment.

Application of this theorem to elements of the set W provides the fundamental result: When v is a continuous quantity whose statistics are described by some probability density function in the set \mathcal{F} , if for all $a \leq b \leq c \leq d$ in the frame of discernment,

$$\min_{f \in \mathcal{F}} \left[\int_a^d f(v) dv \right] \geq \min_{f \in \mathcal{F}} \left[\int_a^d f(v) dv \right] \quad (24)$$

$$+ \min_{f \in \mathcal{F}} \left[\int_a^d f(v) dv \right] - \min_{f \in \mathcal{F}} \left[\int_a^d f(v) dv \right]$$

where the $\min[\cdot]$ operator selects the probability density function in \mathcal{F} that minimizes the definite integral, then for all intervals $(a, b]$ on the frame of discernment,

$$S(a, b) = \min_{f \in \mathcal{F}} \left[\int_a^b f(v) dv \right] u(b-a) \quad (25)$$

where the unit step function $u(\cdot)$ indicates zero support when the lower limit of the interval, a , is greater than the upper limit of the interval, b . While pathological cases can be developed to show that Eq. 24 does not always hold, most practical sets of probability density functions satisfy Eq. 24.

Denote by $f_L(v; a, b)$ the probability density function in \mathcal{F} that minimizes $\int_a^b f(v) dv$.

Although f_L is a function of a and b , it may vary only with v over some range of intervals, so that within this range of intervals,

$$S(a, b) = \int_a^b f_L(v; a, b) u(b-a) dv \quad (26)$$

As the interval changes, however, a different probability density function in \mathcal{F} may apply, and the transition between probability density functions may be either smooth, as in the case of smoothly varying parameter values within a single parametric form, or discontinuous, with a sudden change between parameter values or parametric forms.

When the partial derivative of $f_L(v; x, y)$ is defined with respect to y , and Eq. 24 holds, the mass density function that corresponds to $S(x, y)$ is obtained by equating the integrands in Eq. 21 and Eq. 26 and differentiating with respect to b , then substituting x and y for a and b ,

$$M(x, y) = \left[\frac{\partial}{\partial y} f_L(x; x, y) \right] u(y-x) + f_L(x; x, x) \delta(x-y) \quad (27)$$

The mass density function is zero wherever the same probability density function is used to calculate support in Eq. 25, except along the line x equal to y , where the mass density function is a ridge with value equal to the minimum value of the probability density functions in \mathcal{F} .

When there is no ignorance about the statistical characteristics of the quantity v , the set \mathcal{F} consists of a single element,

$f(x)$, and the mass density function's volume is concentrated on the line x equal to y ,

$$M(x, y) = f(x) \delta(x-y) \quad (28)$$

Equation 13 and Eq. 15 show that, in this case, the support equals the plausibility for all intervals, and the result is a conventional statistical description. When there is ignorance in the statistical description \mathcal{F} , and Eq. 24 holds, Eq. 25 and Eq. 27 provide the means to compute mass density functions from statistical descriptions. Using the equivalence of mass density function nonnegativity and support function superadditivity, however, Eq. 24 need not be verified separately. Rather, the mass density function can be developed without regard for Eq. 24, and if the resulting mass density function is nonnegative, Eq. 24 is guaranteed to hold.

Bounds on Support and Plausibility

Dempster's upper and lower distributions provide bounds and alternative expressions for the support and plausibility. A lower bound on the support function, $S_-(a, b)$, can be derived from the upper and lower distribution functions in Eq. 5 and Eq. 6,

$$S_-(a, b) = \max[0, (F_+(b) - F_+(a))] \quad (29)$$

It is proven in Ref. 9 that

$$S_-(a, b) \leq S(a, b) \quad (30)$$

Moreover, it is also shown that $S_-(a, b)$ satisfies the properties of a support function expressed in Eqs. 9, 10, and 11, and thus can be used in replacement of the exact support function at the cost of adding some ignorance. A corresponding pseudo-mass density function can be derived from $S_-(a, b)$, using the relationships defined in Eq. 13 and Eq. 16. As shown in Ref. 9, the pseudo-mass density function retains all the properties of a mass density function. Interestingly, using the pseudo-mass density function in Eq. 15 yields the plausibility, $P(a, b)$, rather than an upper bound on the plausibility. Hence, the plausibility can be easily calculated from the upper and lower distribution functions

$$P(a, b) = F_+(b) - F_-(a) \quad (31)$$

Example

To illustrate the derivation of evidence theory belief from incomplete statistical descriptions in a data fusion context, consider two sensor platforms monitoring traffic on a highway. A patrol car equipped with a speed measuring radar estimates the speed of each vehicle, and an aircraft using an optical device monitors traffic, estimating vehicle speed with a timing device and distance markings on the pavement. Fusion of reports occurs at a central station in radio contact with the two platforms, using the incomplete data provided. In order to maintain the emphasis on representation of sensor data, the problem of associating ve-

hicles reported by the different platforms is avoided by assuming sparse traffic. Furthermore, the mechanics of the fusion operations are de-emphasized relative to the primary emphasis of representing belief in sensor data to support fusion.

The example is designed to illustrate the differences between fusion of numerical information before extraction of symbolic information, and the fusion of symbolic information separately inferred from the individual information sources. The fusion of numerical information before extraction of symbolic information is not always possible, as shown in the example, but when it is possible, more information is preserved after fusion.

Suppose the specifications for the radar indicate that the random errors in estimating speed are additive, uniformly distributed within ± 4 miles per hour of the calibrated speed, and independent of the actual speed. The radar calibration procedure is guaranteed to place the mean measurement within ± 2 miles per hour of the actual speed, with the calibration bias of the particular unit unknown. Furthermore, errors associated with manually estimating vehicle speed from the aircraft using the highway markings and the timing device are modeled as additive, independent of actual vehicle speed at highway speeds, and dominated by the reflexes of the human. Suppose experiments show that these errors are also uniformly distributed, with constant mean of zero but extent that varies from ± 2 mph to ± 5 mph, depending on the alertness of the human.

If the speed estimate from the patrol car radar is reported as 57 mph, the true speed can be modeled by a set of probability density functions,

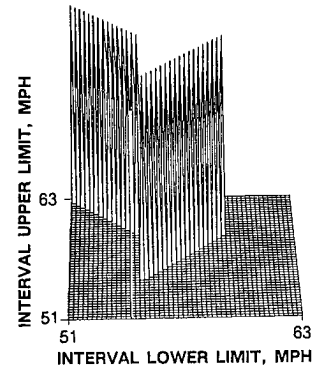
$$\mathcal{F}_1 = \left\{ f(v) : f(v) = \frac{1}{8} [u(v-a-4) - u(v-a+4)], 55 \leq a \leq 59 \right\} \quad (32)$$

while when the speed estimate from the aircraft is reported as 53 mph, the true speed can be modeled by a set of probability density functions,

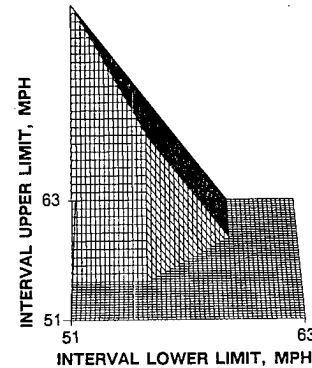
$$\mathcal{F}_2 = \left\{ f(v) : f(v) = \frac{1}{2a} [u(v-53-a) - u(v-53+a)], 2 \leq a \leq 5 \right\} \quad (33)$$

The associated mass density functions, support functions, and plausibility functions for the true speed, based on each individual sensor report and derived from the statistical descriptions outlined in the example description, are shown in Fig. 2 and Fig. 3 respectively, based on derivations in Ref. 9. Figure 2 shows that there is zero support for the proposition that the speed is between 51 mph and 53 mph, since \mathcal{F}_1 contains probability density functions that assign zero probability to this event.

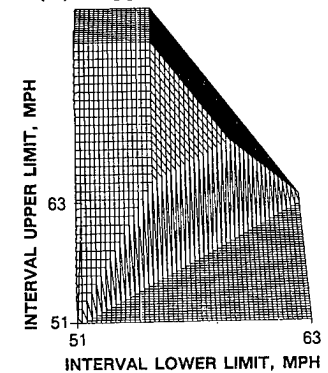
Conversely, plausibility of 0.25 assigned to this event reflects the fact that the largest probability assigned to this event by any probability density function in \mathcal{F}_1 is 0.25. Propositions involving larger intervals and those near the center of the interval containing α tend to have larger support. Fusion of this information using Dempster's Rule for mass density functions presented in Ref. 8 yields a single mass density function that expresses the combined



(a) Mass Density Function



(b) Support Function

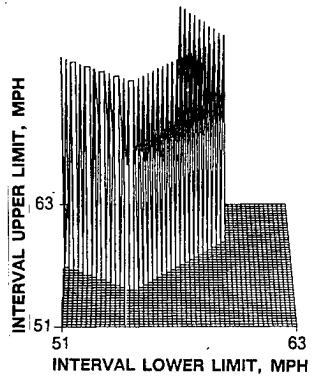


(c) Plausibility Function

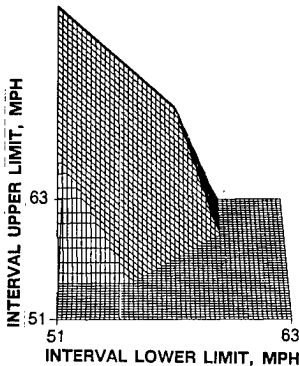
Figure 2 Belief Representations for Speed Estimation from Radar

sitions concerning the speed of the vehicle occupying an interval.

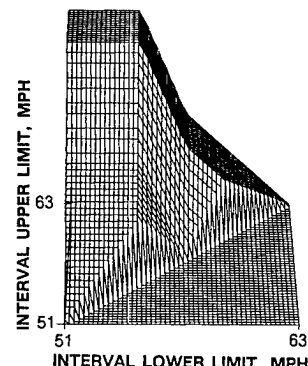
When multiple sensors provide information on different frames of discernment, fusing the numerical information may be more difficult. For example, the sensor might provide information in different coordinate frames, in different units of measure, or at different levels of detail. One approach is to define a frame of discernment on the product space of the multiple sensors. Manipulating mass density functions on a product space



(a) Mass Density Function



(b) Support Function



(c) Plausibility Function

Figure 3 Belief Representations for Speed Estimation from Aircraft

frame of discernment, however, can be prohibitively complicated. Another alternative is to transform the measurements from the multiple sensors onto a common fusion domain, so that the information may be combined in a common frame of discernment. This transformation process may also be complicated, and may also introduce additional uncertainty. Another alternative is to coarsen the frames of discernment to a common symbolic frame, as described in Ref. 2. The information can easily be combined on the symbolic frame using Dempster's Rule. In the highway surveillance example, the observer at the central station may be interested in the type of vehicle, with the frame of discernment consisting of car, van, and truck. If the central station receives information from the patrol car that the vehicle has a commercial license plate that is known never to be assigned to cars, this information assigns a unit of mass to the set {van, truck}. The airborne observer may be able to estimate the length of the vehicle with an optical sensor at known altitude, with vehicle length serving as a discriminant between the three vehicle types. As for the speed estimates previously described, suitable models of the length estimation process can be used to obtain a mass density function for the length estimate.

The separate patrol car and the airborne observer reports providing information on vehicle type cannot be fused directly, since the observations apply to fundamentally different frames of discernment. Both observations can be fused on the frame of discernment describing vehicle type, using known mappings from license plate type and vehicle length to vehicle type. This mapping of length to vehicle type is, in effect, a coarsening of the mass density function to a discrete, symbolic frame of discernment. When this coarsening of a mass density function to a discrete frame of discernment is performed at individual sensors, the fusion process is simplified, at the cost of losing flexibility and detail since the frame of discernment must be predefined.

DISCUSSION

A rigorously justified method for deriving evidence theory belief descriptions from statistical models of sensor performance has been presented. The method is based on the use of mass density functions that restrict the set of focal propositions concerning numerical values to of contiguous intervals. The two-dimensional mass density functions emphasized in the paper, which represent mass assignments to single contiguous intervals, are adequate for many sensor fusion applications. This approach has many attributes, including:

- Formally linking evidence theory and statistical theory representations of uncertainty

- Providing a rigorous way to derive belief from statistical models
- Offering an alternative, more intuitive way for experts to describe uncertainty in the sensor outputs when rigorous derivations are not used.

The method is based on the seminal work of Dempster, integrated with Shafer's emphasis on the properties of belief functions, and Strat's elegant representation of mass assignments to numerical quantities.

A simple example shows how both low-level fusion of numerical information and extraction of belief in symbolic information to support high-level fusion are supported by this approach. Low-level fusion preserves mass assignments over the entire original frame of discernment, allowing more flexibility in extracting symbolic information from the fusion product. High-level fusion of symbolic information extracted from the measurements of individual sensors is equivalent to fusion on coarsenings of the frames of discernment associated with the original mass density functions. When two sensors provide inherently different information, for example in different coordinate systems, or when one source of information is inherently symbolic, the disparate types of information from the different sensors must be mapped to a common frame of discernment before fusion. The mapping is generally simplest when the frame of discernment used for fusion involves discrete, symbolic propositions.

This approach to deriving belief in numerical quantities represents a contribution towards the fusion of numerical information using Evidence Theory, but not a completed solution to the practical problems of representing belief in numerical quantities. While providing a mechanism for translating incomplete statistical descriptions into belief measures, it does not address the derivation of the statistical descriptions. In some cases, as in the examples presented, these are a natural and straightforward way to acquire the requisite statistical information. In some other applications, there is considerably more difficulty in deriving a set of density functions to describe the numerical quantity of interest. In the cases when analytical formulations may need to be supplemented by heuristics, expression of heuristic information in terms of probability density functions may be a more natural and richer representation for sensor experts, rather than forcing experts who

generally think in terms of statistical models to describe their belief directly in terms of mass assignments on a continuous or discrete frame of discernment. Therefore, this approach has applications for both rigorous and heuristic representations of belief in numerical quantities.

ACKNOWLEDGMENTS

Martin G. Bello contributed to the underlying mathematical developments.

This research is supported by the Air Force Systems Command, Aeronautical Systems Division, Under Contract Number F33615-87-C-1404.

REFERENCES

1. A.P. Dempster, "Upper and Lower Probabilities Induced by a Multivalued Mapping," *Annals of Mathematical Statistics*, 38:325-339 (1967).
2. G. Shafer, "A Mathematical Theory of Evidence," University Press, Princeton (1976).
3. G. Shafer, "Belief Functions and Parametric Models," *Journal of the Royal Statistical Society B*, 44:322-352 (1982).
4. G. Shafer, "Probability Judgment in Artificial Intelligence and Expert Systems," *Statistical Science*, 2:3-44 (1987).
5. T.D. Garvey, et al, "An Inference Technique for Integrating Knowledge from Disparate Sources," *Proceedings of the Seventh Joint Conference on Artificial Intelligence*, 253-255 (1981).
6. I. Kadar, "Data Fusion By Perceptual Reasoning and Prediction," *Proceedings 1987 Data Fusion Symposium*, 217-224 (1987).
7. R.A. Hummel and M.S. Landy, "Statistical Viewpoint on the Theory of Evidence," *IEEE Transactions in Pattern Analysis and Machine Intelligence*, 10:235-247 (1988).
8. T.M. Strat, "Continuous Belief Functions for Evidential Reasoning," *Proceedings of the American Association for Artificial Intelligence*, 308-313 (1984).
9. R.W. Pinto et al, "Sensor Algorithm Research Expert System Model-Based Vision Architecture," *The Analytic Sciences Corporation, Technical Report TR-5417-1* (1987).

THIS PAGE LEFT BLANK INTENTIONALLY

PROACTIVE DATA FUSION FOR INTEGRATED TACTICAL WARNING AND ASSESSMENT*

Joseph G. Wohl

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, MA 01803

ABSTRACT

A study of the CINCNORAD strategic attack assessment and warning function was performed to determine whether a proactive approach to sensor/intelligence fusion could significantly reduce the time required to reach a high-confidence assessment. A comparative example was analyzed to evaluate the potential benefits and costs of proactive fusion. The example used a complex, calendarized Soviet test launch scenario requiring discrimination among various situation alternatives. The major findings were that:

- Post-event diagnosis requires multiple information requests and takes time. Processing is done in spurts based on requests.
- The proactive method saves critical steps and time but requires continuous processing effort and additional resources.
- Much of the required processing was already being done within NORAD, but under post-event, high-stress conditions; and the critical information was not adequately organized or shared among the key NORAD information centers.
- The different centers within NORAD must learn to anticipate each others information needs as critical situations develop.

* Study sponsored by U.S. Air Force, Armstrong Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Ohio under subcontract to Systems Development Corporation.

PROACTIVE DATA FUSION FOR INTEGRATED TACTICAL WARNING AND ASSESSMENT*

Joseph G. Wohl

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, MA 01803

INTRODUCTION AND SUMMARY

In 1985, the Air Force Armstrong Aerospace Medical Research Laboratory sponsored a study of NORAD's Integrated Warning and Attack Assessment (ITW&A) process. The purpose of the study was to determine whether a proactive approach to the fusing of intelligence and other information with sensor data could significantly reduce the time required to reach a high-confidence assessment. The results were positive, and were briefed to representatives of the Air Force Space Command.

ITW&A DECISIONMAKING PROCESS

Responsibility for the ITW&A decision process and its products lies with CINCNORAD. Other commands such as the NCA and CINCSAC also involve key decisionmakers to whom the results of the ITW&A process represent time-critical inputs. In brief, CINCNORAD monitors the developing strategic situation, tries to reduce uncertainty, and decides what his report to the other commands will be. He must also manage the widespread information system which brings him the requisite information on which to base his decisions.

As a matter of course, this information system must provide CINCNORAD with an accurate and timely perception of the developing situation. But it should also help him to "walk through" critical processing and decision sequences; to keep track of situation changes; to understand the meaning of developing information patterns by helping him to create and evaluate hypotheses about their meaning; and finally, to help in selecting appropriate responses.

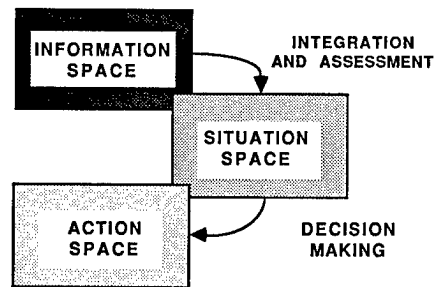
On the other hand, CINCNORAD emphatically does not want an information system which he must fight to get what he needs when he needs it -- it must be "user-friendly" and extremely responsive to his needs. Nor does he want an information system which tries to make his assessments and decisions for him, either implicitly or explicitly.

THE ITW&A INFORMATION SYSTEM

The ITW&A information system consists of all of the sensors (both space- and ground-based), communication links, intelligence sources, world-wide weather data sources, etc., together with the personnel and organization of the CINCNORAD staff. The CINC must utilize this system to make time-critical assessments and decisions regarding missile and air attacks against both the continental United States and its allies.

* Study sponsored by U.S. Air Force, Armstrong Aerospace Medical Research Laboratory, Wright-Patterson Air Force Base, Ohio under subcontract to Systems Development Corporation.

From the perspective of control and estimation theory, there appear to be three separate "state spaces" involved in the CINC's decisionmaking activities. As depicted in Fig. 1, they are: (1) the information space, which is large and continually changing; (2) the situation space, which is small and represents all feasible correlated patterns (i.e., possible hypotheses or meanings) of the data in the information space; and (3) the action space, which represents the feasible actions which the CINC can take and therefore is highly constrained. The integration and assessment process serves to map the information space onto the situation space, while the decision process serves to map the situation space onto the action space (in each case, a many-to-few mapping). These processes require that all of the major centers in NORAD share key information among themselves in order to provide the needed context for reducing both assessment uncertainty and response time.



R-4674

Figure 1. The ITW&A Information System.

Assume for example that CINCNORAD receives a critical event report including event type, location, and time; and that this report could signify one of two possible situations:

- (1) A real attack is imminent or already taking place. In this case the correct action is to transmit an attack warning and alert all strategic forces.
- (2) No attack is imminent or taking place -- the perceived event is due to accident, storm, miscommunication, exercise, foreign or domestic test launch, etc. In this case the correct action is simply to report the situation and insure that no other commands may be mistaking it for something more serious.

Given the information input, the two possible meanings are paralleled by two possible types of errors:

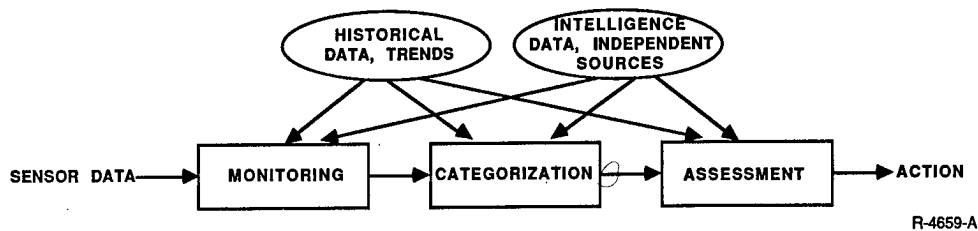


Figure 2. Fusion: The Data Integration and Assessment Process.

- (1) False alarm, resulting in alerted forces even though no attack is taking place.
- (2) False dismissal, resulting in delayed alert even though a real attack is taking place.

Clearly, the cost of a false dismissal can far overshadow the cost of temporarily alerting forces; on the other hand, a continuing sequence of false alarms can result in decreased decision sensitivity, as occurred soon after the DEWLine was originally activated.

Thus, discrimination among alternative situation perceptions is critical, and the ITW&A system must provide the capability for improving this discrimination, i.e., for reducing uncertainty.

FUSION: THE DATA INTEGRATION AND ASSESSMENT PROCESS

As noted by Wohl (1981), the process of military decision-making is characterized by two distinct types of uncertainty:

- Situation uncertainty ("What is the situation?")
- Action uncertainty ("What should be done about it?")

Situation uncertainty is associated with the mapping of information space into situation space, while action uncertainty is associated with the mapping of situation space into action space. The focus in this paper is on situation uncertainty, since one of CINCNORAD's primary responsibilities is that of accurate and timely attack assessment, warning, and characterization. The process by which situation uncertainty is reduced has been termed "Fusion" (sometimes "Ops/Intel Fusion") by the military, by which is meant the integration and assessment of sensor, intelligence, and other relevant data in order to provide a clear and complete situation picture with uncertainty minimized to the greatest possible extent. At NORAD, at the time of this study, the process was called Integration and Assessment.

Figure 2 summarizes the fusion or data Integration and Assessment (I&A) process. It involves three major activities: Monitoring, Categorization, and Assessment. Sensor data is shown as the primary input to the Monitoring process, which involves the blocking or filtering out of unwanted or irrelevant data (e.g., the moon or flocks of birds as targets), as well as aggregation of data from several sensors. Validation of the sensor data is also performed here.

The filtered, aggregated and validated data then enters the Categorization process, which involves correlation of multi-source data (e.g., track correlation from multiple radars) and interpretation (e.g., test launch, ripple attack). Finally, the Assessment process involves the integration of all available data; the formulation of a single succinct situation summary; and the final confirmation of that summary prior to taking appropriate action. As indicated in Fig. 2, each of these three processes also makes use of data from past history, independent sources, and trend data, as well as from intelligence sources.

The over-all I&A process shown in Fig. 2 is what maps the information space onto the situation space in Fig. 1. But confidence in the accuracy of the mapping requires that questions be asked to test that mapping, which is itself a hypothesis about the

situation. The kinds of questions which trained decisionmakers ask generally take the form of hypothesis tests to reduce situation uncertainty:

If hypothesis A is correct, then we should be observing the occurrence of certain other related and predictable events. Are they being observed?

Experience with decisionmakers and their staffs in high-level headquarters indicates that they may ask a series of questions about almost anything, sometimes (and often to the consternation of the staff) with no apparent relation to the problem at hand. However, careful examination of the situation faced by the decisionmaker at a given time vis-a-vis the questions being asked at that time strongly suggests that they are testing their existing situation perception and examining alternative hypotheses. This "Integration and Assessment" (I&A) Refinement Loop" is shown in Fig. 3.

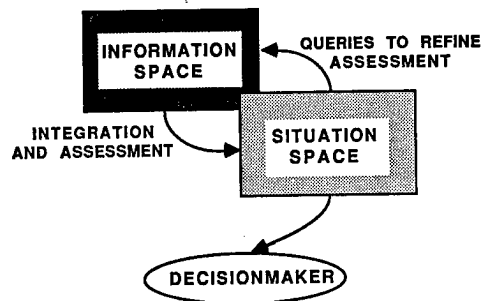


Figure 3. The Integration and Assessment Refinement Loop.

A unique example of how the I&A Refinement Loop operates is shown in Fig. 4. Suppose that four alternative hypotheses (A, B, C, and D) are supportable given the data at hand (the location marked by an "x" represents ground truth). The size of the circles represents the degree of uncertainty associated with each hypothesis. Note that by reducing the uncertainty associated with hypotheses B, C, and D, we are left with hypothesis A by simple process of elimination. Further reduction of uncertainty regarding hypothesis A is unnecessary. Many military situations are of this type and can be resolved by "asking the right questions."

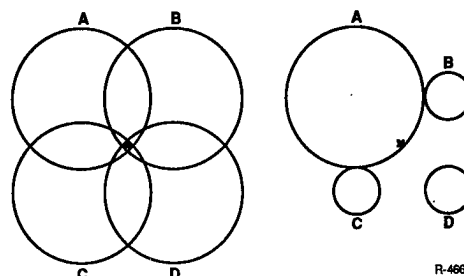


Figure 4. Results of Refinement Process.

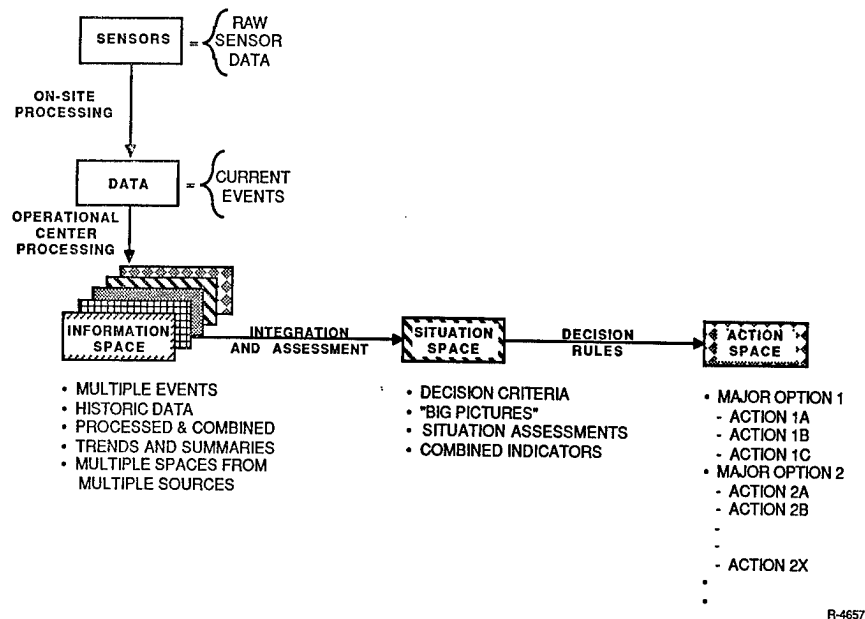


Figure 5. Current I&A Method: Diagnostic.

DIAGNOSTIC METHOD OF INTEGRATION AND ASSESSMENT

The I&A method in use at the time of this study (1985) is depicted in Fig. 5. Because most of the processing activity occurs after detection and recognition of the initial events, the term "diagnostic" is used to describe this process. In other words, the events are considered as symptoms, and the purpose of I&A is to reliably associate a most likely cause with these symptoms. The generation and testing of alternative hypotheses concerning possible causes is the most critical of the diagnostic activities.

The relatively short time of flight of enemy ICBMs and SLBMs requires that the diagnostic process be performed under high time-stress conditions. Raw sensor data is first pre-processed at the sensor sites (or at the Ground Entry Points for satellite data) to eliminate sensor false alarms. This pre-processed data then enters the NORAD Space Defense Operations Center (SPADOC). Here, data from multiple sensors are combined and correlated to increase confidence in event detection and minimize the incidence of misjudgments of event type. This processed data now becomes part of the "Information Space" previously described. The initial events serve to trigger requests for relevant information, which in turn spawn additional processing activity in SPADOC to further correlate or match the event data with information received in other NORAD operational centers (Air Defense, Intelligence, Weather, etc.). Thus, the purpose of this processing activity is to sharpen the situation perception and to reduce the likelihood of errors both in identifying event patterns and in discriminating among their possible interpretations as shown in Table 1.

Table 1. Possible Interpretations of Sensor Data
- Non-threatening enemy action
- Natural phenomenon
- Unusual condition of own system
- Third-world or terrorist attack
- Limited attack
- Saturation attack
- etc.

PROPOSED PROACTIVE METHOD OF INTEGRATION AND ASSESSMENT

The proposed approach is to perform most of the diagnostic activities continuously, before any detection or event reports are ever received. Using the same historical data, independent source data, trend data, and intelligence data, the various NORAD centers (the Intelligence Center, the Air Defense Operations Center, the Missile Warning Center, etc.) would be tasked to continuously look for patterns and trends in their individual information sets which might be associated with specific developing situations.

In the proposed proactive method, the received data must be considered to be the symptoms; the purpose of the I&A process is to reliably associate several possible causes with these symptoms, before any event occurrence. The detection and recognition of one or more events can then serve as an immediate discriminant among alternative situation perceptions or hypotheses, or as confirmation of a highly likely situation. The many critical diagnostic activities previously performed on request under high time-stress conditions are now performed continuously as a matter of course, and prior to event detection. Thus, the proactive method serves to greatly "stretch" the time available for hypothesis generation and testing.

COMPARATIVE EXAMPLE: SOVIET TEST LAUNCH

Table 2 has been constructed to aid in exploring the differences between the diagnostic and proactive methods of integration and assessment. Note the fundamental difference between the sequences in Columns A and B. The "Diagnostic" sequence in Column A is triggered by the occurrence of the launch event in step 1, resulting in the immediate convening of the "Beige loop" conference (step 2) among key decision personnel in the various Centers within NORAD. Steps 3 through 7 must then intervene before the Soviet test launch is confirmed and the ITW&A product is disseminated in steps 8 and 9.

By way of contrast, the "Proactive" sequence in Column B shows how continual hypothesis generation activity (based on intelligence and other data), taking place prior to launch detection, can result in significantly reduced time between launch detection and dissemination of the ITW&A product, as in steps 8 through 11.

Table 2. Comparative Example: Soviet Test Launch

A. Diagnostic Method	B. Proactive Method
1. Launch event detected, recognized and validated	1. Intel sources notify re: activity buildup at known Soviet test launch site (5 to 10 day advance notice of test launch)
2. "Beige loop" conference is convened	2. Intel sources notify re: departure of Soviet tracking and telemetry ship from home port (3 day advance notice of test launch)
3. Independent sources identify presence of Soviet tracking and telemetry ship at known sea location; also report Soviet ocean closures	3. Independent sources notify re: ocean closures plus arrival of tracking and telemetry ship at known sea location
4. Review of past history shows presence of same ship at same location during four previous test launches from this site	4. Review of past history indicates presence of same ship at same location whenever launches occur from this test launch site
5. Review of intel data shows activity buildup at test launch site during previous week	5. Prediction of launch site weather for next 48 hours further narrows event time
6. No supporting indications can be found for alternative hypotheses	6. Alert sensor stations to probable test launch within 24 hours
7. Event interpreted as Soviet test launch, based on ocean closures; launch location, time, and type; and trajectory	7. No supporting indications found for alternative hypotheses
8. Test launch is confirmed	8. Launch event detected, recognized and validated
9. Appropriate ITW&A product is disseminated	9. Conference is convened
	10. Test launch is confirmed
	11. Appropriate ITW&A product is disseminated

CONCLUSIONS

It is clear that post-event diagnosis is done in "spurts." It is based on mutual requests for information, which must take place under high-stress conditions. This takes time, and in addition, requires multiple information requests and coordination among the information centers in NORAD.

The proposed proactive method is not without cost, however. While it saves critical time, it requires additional and continuous processing effort. Even though most of this effort can take place well before launch detection, it clearly would require additional resources as well as changes in the way in which intelligence data is handled and communicated among the NORAD centers. Also, the patterned movement of the Soviet tracking ship might well be a "cover" for a hostile ASAT launch, so that the additional processing effort must include a specific hypothesis to be tested for just such an instance.

It is interesting to note that much of the required processing for the proactive approach was already being done at NORAD at the time of this study, but the information was not adequately organized within the NORAD centers, nor was it being appropriately shared among the centers. Finally, it must be remembered that the most critical processing was being carried out under post-event, high-stress conditions.

In any case, it is also clear that the various NORAD centers must learn to anticipate each others information needs as strategic threat situations develop, before any launch events occur and the CINC gets involved.

One final point is worth making with regard to the implications of the diagnostic vs. proactive methods for the Strategic Defense Initiative (SDI) program. The Weapons Enable decision will perhaps be the most critical single command and control decision for which humans must take responsibility in an operational Space Defense system. The various SDI architecture studies have repeatedly demonstrated the severe impact of decision delays on system effectiveness (i.e., on reentry vehicle leakage rates). For this reason, proactive data fusion will more than likely be an absolute necessity for an effective Space Defense system.

REFERENCE

- [1] Wohl, J.G., "Force Management Decision Requirements for Air Force Tactical Command and Control," IEEE Transactions Systems, Man, and Cybernetics, SMC 11:9 (Sept. 1981).

THIS PAGE LEFT BLANK INTENTIONALLY

MANAGING TEMPORAL UNCERTAINTY IN SITUATION ASSESSMENT

David Noble

Engineering Research Associates
1595 Springhill Road
Vienna, Va. 22180

A template system, which incorporates time-event models of hostile activities, helps analysts infer hostile objectives and plans from a pattern of observables. This system represents and manages uncertainties about the identity and time of events, the relationships among events, and the event participants. Uncertainties are represented as different types of intervals. Inference and evidence combination are accomplished through operations on these intervals.

The uncertain time of an event is represented by two intervals: a broader interval representing the full range of possible times of the event, and a narrower interval representing the typical times for the event. The former interval is highly reliable but often very large. The latter interval, which is based on expert judgment of what usually occurs, is often much more precise but much less dependable.

Times for events are estimated from the times of reported activities and template models of hostile operations. Each template model describes the events in an operation, the range of typical and possible durations for each of these events, and the ranges of typical and possible time intervals between events.

There are two steps in computing the estimated time of a future event from the estimated time of an earlier event. In the first step, the time of the future event is projected from the earlier one using the duration and time interval information in the template. In the second step this projected interval is combined with the prior estimate of the future event to produce a refined estimate.

The computational methods for estimating the "possible" and "typical" times differ in both of these steps. The estimate of possible times uses bounding techniques while the estimate of typical times approximates probability methods.

In projecting the earliest possible time of a future event, the earliest start time of the first event is added to the shortest time interval between the earlier and later event. The latest start time is computed similarly. When refining this estimate by combining the estimate from two sources, the estimates from each source are treated as completely reliable. The earliest possible start time is the later of the two estimates of earliest start time in each source, and the latest possible start time is the earlier of these estimates.

In projecting the earliest and latest typical start times of a future event, the earliest and latest start times of the earlier event and the earliest and latest likely time intervals between events are considered to reflect the mean and standard deviations of two independent normal distributions. The earliest and latest start time of the future event reflect the sum of these distributions. When refining this estimate by combining evidence from two sources, each source is considered to provide independent estimates of typical times, based on different data and considerations. An event time in the refined estimate is considered to be within the typical interval if at least one source of evidence considers the time to be typical, and the other considers it to be typical or close to typical.

MANAGING TEMPORAL UNCERTAINTY IN SITUATION ASSESSMENT

David Noble

Engineering Research Associates
1595 Springhill Road
Vienna, Va. 22180

INTRODUCTION

Template-based situation assessment infers possible ongoing hostile operations from a pattern of reported observables. Each template specifies the observations expected given that a particular hostile operation is being conducted. An operation associated with a template is inferred when enough of the observations specified by the template are reported.

One of the major difficulties in developing workable template-based systems is defining templates that can accommodate operational variability. While it may be easy to develop a template that describes one particular "typical" hostile operation, it is much more difficult to develop more flexible templates able to describe a range of possible operations. Such flexibility is required in a templating system, for very few hostile operations are likely to occur in an exactly "typical" fashion. A template unable to recognize these less typical operations is unlikely to be very useful.

The approach used here for accommodating operational variability is motivated by models of human memory organization and information processing (Ref 1). This model has led to templates that represent data in the kind of "flexible way that reflect human tolerance for vagueness, imprecision, and quasi-inconsistencies" (Ref. 2).

This flexibility and tolerance for imprecision arises from the way that uncertainty is represented within the templating system. There are many different types of uncertainty in a template: uncertainty about the type of operational events, about the times of these events, and about the types, numbers, missions, and locations of hostile platforms. Each of these must be represented and managed.

This paper describes how one of these types of uncertainty, temporal uncertainty, is handled. It describes how different types of temporal uncertainty are represented, how uncertain times are projected, and how

different types of evidence about the times of events are fused.

The following example introduces the principal concepts in managing uncertainty in the templating system. It shows the kinds of inferences about the times of events that are made by the templating system. The method for computing these times is described in the sections following this example.

AN OPERATIONAL EXAMPLE

In this example, a U.S. force in the Indian Ocean is tasked to evacuate U. S. citizens from Pakistan. Intelligence indicates a possible Soviet air strike against the carrier. A Soviet reconnaissance flight is reported to be in progress. Its take off time is not certain, but was probably between 0830 and 0840, and was certainly not before 0800 or after 0900. We have been asked to appraise the commander of the progress of this possible strike, and to advise him as early as possible when the launch may occur.

Figure 1 illustrates our estimate of the times of some of the key events. These events are the take off of the reconnaissance aircraft, the time that this aircraft will likely find us, the time that the strike aircraft will arrive in the area patrolled by our combat air patrol (CAP), and the time that the launch will occur.

None of these times can be estimated exactly, but each can be estimated approximately. Figure 1 represents these estimated times as two time intervals. The outer interval, colored gray, represents the range of possible times. The inner interval, colored black, represents the most likely times. The intervals for "take off" reflects our recently received report.

This hybrid representation of possible and likely times is intended to combine the advantages of estimates based on capability with the advantages of estimates based on hostile doctrine and tactics. The estimates

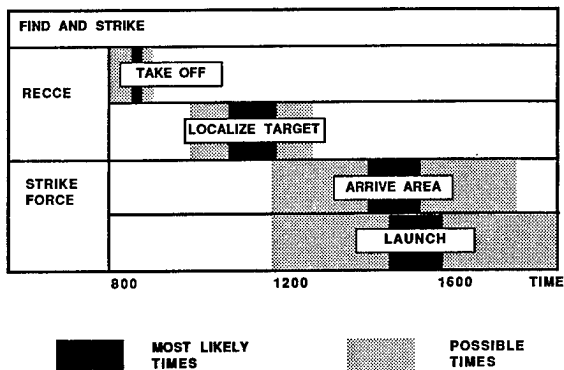


Figure 1. Estimated times of hostile events based on reported take off.

of possible times, which are based on hard intelligence data about hostile forces, are highly reliable. In Figure 1 the bounds on possible times are based on fuel capacity, nearest and farthest possible launch points, and maximum aircraft speed. As suggested by the figure, these estimates tend not to be very constraining, permitting a range which may be very large.

The estimates of likely times are based on typical tactics, likely aircraft launch locations, typical types of aircraft, and average luck in finding the U.S. forces. As suggested in Figure 1, the bounds on typical times can be much narrower than the bounds on the possible times. Unfortunately, there is no guarantee that the actual time of the event will fall within these bounds, for hostile forces are free to conduct an operation in an atypical way.

These estimates illustrate inferences that may be made from a single contact report of the probable and possible take off times for a reconnaissance aircraft, and from knowledge about hostile capabilities, doctrine, and tactics.

Initially, the estimates made from such reports may be very uncertain. As additional reports are received, however, they can often be considerably refined. In our example, we assume that the following reports will be received:

Reported Activity	Time of Event
Reconnaissance take off	Possible: 0800 to 0900 Probable: 0830 to 1040
Our force detected	Possible: 1040 to 1140 Probable: 1120 to 1130
Strike force deployed	Possible: 1140 to 1210 Probable: 1140 to 1200
Strike force in area	At 1450
Missile launched	At 1522

Figure 2 illustrates how the estimated time of the hostile missile launch can be continuously refined after each of these reports. After the initial reconnaissance report, it is estimated that the launch will likely occur between 1800 and 1900. It is

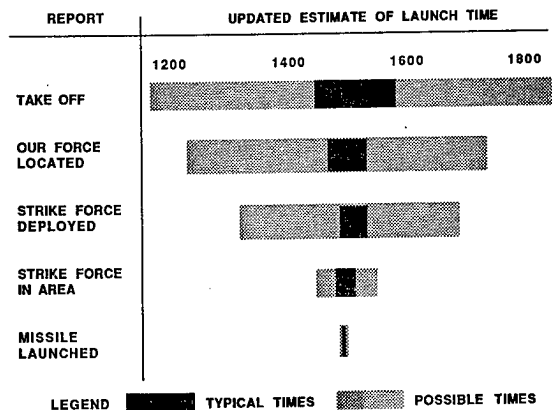


Figure 2. Estimated launch time after receiving each report.

physically possible, however, for the launch to occur much earlier or much later. The latest time assumes that the launch will be delayed until the last possible moment, limited only by the fuel capacity of the targeting and strike aircraft. The earliest time assumes that the reconnaissance aircraft is very lucky finding us, that the fastest strike aircraft will be used and will take off immediately.

Each of the reports about the hostile forces conveys information about the progress of the attack, and each contains information enabling the estimated launch time to be refined. Note that in all cases, until the launch itself is detected, the range of possible times is significantly broader than the range of typical times.

TEMPLATE REPRESENTATION OF EXPERT KNOWLEDGE

In the example the estimates of event times were based on reports of hostile activity and an understanding of hostile force capabilities, doctrine, and tactics, and were computed using mathematical methods for managing uncertain times. The understanding about hostile forces required for these calculations is contained within reference templates described immediately below. The mathematics is described in the following section.

A reference template is a time-event model that describes a type of military operation. Each reference template models an entire class of operations, rather than just a single specific one. Each describes not only the typical activities that comprise the operation, but also the likely and permissible variations in these activities. Templates used for military situation assessment also specify the indicators and observables associated with the operation. Reference templates are developed by operational experts, and capture expert knowledge about hostile tactics and doctrine.

Figure 3 summarizes the reference templates used in the preceding example. The "Find and Strike" template describes the overall attack. The four small templates at the bottom of the figure describe the main

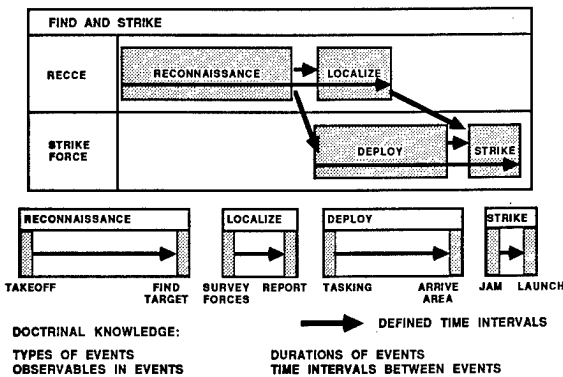


Figure 3. Template representation of hostile capabilities, doctrine, and tactics.

events in the attack. In each of the templates the rows denote the major participants, and time runs along the horizontal axis. Important events are depicted on the row of the participant performing them, placed along the horizontal axis at the time that they typically occur.

The "Find and Strike" template has two rows. The top row depicts the activities of the reconnaissance platform. It is responsible for two events, reconnaissance, and localization. In reconnaissance the aircraft finds the U. S. forces. During localization, it tracks the priority targets. The second row depicts the activities of the strike force. Deployment is transit to the target area. The strike entails penetration of our defenses and launch and guidance of the missile.

There are two different types of times specified in reference templates: times between events and event durations.

The arrows in Figure 3 indicate those event pairs that have explicitly defined temporal relationships with each other. These relationships may specify the time interval between the beginnings of two events, between the ends of two events, or between the beginning of one event and the end of another. In Figure 3 there is a time relationship between the end of "Reconnaissance" and the beginning of "Localize". This relationship indicates that localization cannot begin until the U. S. forces have been found and reconnaissance is over. There is also a defined relationship between the beginning of reconnaissance and the end of localization. This relationship limits the total time of these two events to the fuel capacity of the reconnaissance aircraft.

Each of these time intervals may be uncertain. Like the uncertain times for the hypothesized events of Figure 1, the time intervals are also represented by two intervals: an interval showing the bounds on what is possible, and an interval showing the bounds on what is most likely or typical. These two intervals are represented by four numbers. For example the temporal relationship between the end of event "A" (Reconnaissance) and the beginning of event

"B" (Localize) is represented by the four tuple

Time Interval AB = (AB₁, AB₂, AB₃, AB₄)
where

AB₁ = shortest possible time interval
AB₂ = shortest likely time interval
AB₃ = longest likely time interval
AB₄ = longest possible time interval

The duration of events may also be uncertain. Each duration is represented by four numbers denoting the possible and typical time intervals between the beginning and end of the event. These intervals are defined in the detail templates for the events. In Figure 3, for example, the duration of the event "Reconnaissance" in the "Find and Strike" template is defined as the uncertain time interval between the events "Take off" and "Find Target" in the template "Reconnaissance" shown at the bottom of the figure.

The hybrid representation of uncertainty for times between events and for event durations has the same advantage as the hybrid representation for the estimated times of ongoing events. The outer range, the range of possible times, is highly reliable but may often be extremely broad. The inner times, the range of likely times, is considerably less reliable, but is often much more precise.

In illustrations of templates, event duration is suggested by the length of the boxes representing each event.

CALCULATING THE UNCERTAIN TIME OF EVENTS

Uncertain event times, as shown in Figure 1, are represented by four numbers. For example, the time that event "A" (Take off) begins is represented by the numbers

Time A = (A₁, A₂, A₃, A₄) where

A₁ = earliest possible start time
A₂ = earliest likely start time
A₃ = latest likely start time
A₄ = latest possible start time

The uncertainty ranges in the times of a future event are computed from the uncertainty ranges of an earlier event in two steps. In the first step, an estimate of the time of the second event is computed by adding the uncertain time interval between these events to the uncertain time of the first event. In the second step, this projected time is combined with the prior estimate of the time of the second event to produce a refined estimate of the event's time. These steps are central to data fusion, for they combine the evidence about the time of the earlier event with the evidence about the time of the second event to produce a better estimate of the second event time.

Step 1: Projecting Future Times

The bounds for possible and typical projected times are calculated using separate formulae.

Formula for the bounds on possible times

$B_{1fromA} = A_1 + AB_1$ Earliest possible time for B

$B_{4fromA} = A_4 + AB_4$ Latest possible time for B

Formula for the bounds on typical times

Let $U = .5 * (A_2 + A_3 + AB_2 + AB_3)$

Let $S = .25 * (A_2 - A_3)^2 + (AB_2 - AB_3)^2$

$B_{2fromA} = U - S$ Earliest likely time for B

$B_{3fromA} = U + S$ Latest likely time for B

The bounds on the possible intervals, B_{1fromA} and B_{4fromA} are outer estimates of theoretically possible times. The earliest possible time of B, as estimated only from A and the relationship between A and B, is the sum of the earliest possible time of A and the shortest possible time interval between A and B. Similarly, the latest possible time of B is the sum of the latest possible time of A and the longest time interval between A and B.

The bounds on the typical intervals, B_{2fromA} and B_{3fromA} , are calculated differently. Because the concept of "typical" resembles the concept of "probable", the formula for these bounds are based on probability theory, and the projected times behave as expected from operations on probabilities. For instance, a typical interval which is the sum of two other typical intervals is somewhat larger than either of the two contributing intervals.

The formula for the typical interval treats the inner bounds of the uncertain times, A_2 and A_3 , and the inner bounds of uncertain interval between events, AB_2 and AB_3 , as the standard deviation departures from the mean of normal distributions. The inner bounds of the projected time, B_{2fromA} and B_{3fromA} , is interpreted as the standard deviation departure from the mean of the sum of the two normal distributions.

Data Fusion: Combining Estimates of an Event's Time

The refined estimate of a time for an event "B" is computed by combining the estimates of the event time projected from earlier events with the prior estimate of this event time. This process is summarized in Figure 4.

The new estimate for time B_1 , the earliest time that B can begin, is the minimum of the times B_{1fromA} and B_{1prior} , where B_{1prior} is the prior estimate of the minimum start time of event B. This rule reflects the premise, true within each hypothesis though not necessarily true outside the hypothesis, that both estimates, B_{1fromA} and B_{1prior} , are completely reliable. Consequently, B_{1fromA} and B_{1prior} both provide absolute lower bounds. The new lower bound is the greater of these two numbers.

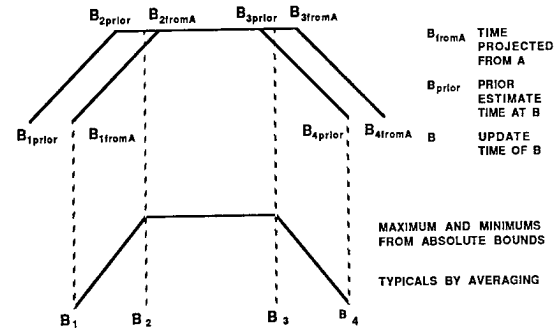


Figure 4. Combination of evidence, refining an estimate of event time.

Similarly, the estimate of the greatest possible time, B_4 , is the minimum of the two estimates, B_{4fromA} and B_{4prior} .

The range of typical values, B_2 to B_3 , is set to those values classified as typical or nearly typical in both contributing estimates. A particular time for B will be regarded as typical if it is regarded as typical in both B_{prior} and B_{fromA} . A time can also be considered to be typical for event B if it is regarded as typical in either B_{prior} or B_{2fromA} and is not regarded as atypical in either estimate. "Not atypical" is a parameter of the program, corresponding in Figure 4 to a height on the descending slope of the uncertainty representation.

SUMMARY

The estimated times of events in a situation assessment are often uncertain. These uncertainties may be represented by two intervals: a broader interval representing the full range of possible times of the event, and a narrower interval representing the typical times for the event. The former interval is highly reliable but often very large. The latter interval, which is based on expert judgment of what usually occurs, is often much more precise but much less dependable.

Times for events are estimated from the times of reported activities and a template model of hostile operations. The template model describes the events in an operation, the ranges of typical and possible durations for these events, and the range of typical and possible time intervals between events.

There are two steps in computing the estimated time of a future event from the estimated time of an earlier event. In the first step, the time of the future event is projected from the earlier one using the duration and time interval information in the template. In the second step this projected interval is combined with the prior estimate of the future event to produce a refined estimate.

The computational methods for estimating the "possible" and "typical" times differ. The estimate of possible times uses bounding

techniques while the estimate of typical times approximates probability methods.

These methods of representing and managing temporal uncertainty are embedded within a system for inferring possible hostile actions from a pattern of observed activities. In the near future the system will represent and manage uncertainties about platforms identities and locations. The system will represent these uncertainties in the same way that time uncertainty is represented, as a range of typical values and a range of possible values.

REFERENCE

1. D. Noble. "Template-Based Data Fusion for Situation Assessment", Proc. 1987 Tri-Service Data Fusion Symposium, vol 1, pp. 156-161.

2.. D. Rumelhart and A. Ortony, "The Representation of Knowledge in Memory", in: "Schooling and the Acquisition of Knowledge", R. Anderson, R. Spiro, and U. Montague, eds., Earlbaum, Hillsdale, N.J. (1977).

Sensor and Information Fusion From Knowledge-Based Constraints

Allen R. Hanson, Edward M. Riseman
Computer & Information Science Department
University of Massachusetts¹
Amherst, MA 01003

Thomas D. Williams
Amerinex Artificial Intelligence, Inc.
274 N. Pleasant Street
Amherst, MA 01002

ABSTRACT

A constraint-based approach to uniformly combining information from multiple representations and sources of sensory data is described. The approach is important to research in intermediate grouping, knowledge-based model matching, and information fusion. The techniques presented extend the capabilities of an earlier system that applied constraints to attributes of single types of extracted image events called tokens. Relational measures are defined between symbolic tokens so that sets of tokens across representations can be selected and grouped on the basis of constraint functions applied to these relational measures.

Since typical low-level representations involve hundreds or thousands of tokens in each representation, even binary relational measures can involve very large numbers of token pairs. Control strategies for ordering and filtering tokens, based upon constraints on token attributes and token relationships, can be formed to reduce the computation involved in producing token aggregations. The system is demonstrated using region and line data and an associated set of relational measures. The approach can be naturally extended to include tokens extracted from motion, stereo, and range data.

¹This work has been supported by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. N00014-82-K-0464, Air Force Office of Scientific Research under contract F49620-83-C-0099, and the National Science Foundation under contract DCR-8500332.

Sensor and Information Fusion From Knowledge-Based Constraints

Allen R. Hanson, Edward M. Riseman
Computer & Information Science Department
University of Massachusetts¹
Amherst, MA 01003

Thomas D. Williams
Amerinex Artificial Intelligence, Inc.
274 N. Pleasant Street
Amherst, MA 01002

1 Introduction

A major problem confronting vision systems which use multiple sensors, or which generate multiple low-level descriptions from image data, is the coherent and consistent integration of information contained in the multiple representations. Most vision systems utilize only one type of sensory data (e.g., visible light, SAR, IR, range) and only one type of low-level process producing a single type of extracted image event, e.g., regions of a region segmentation. However, after many years of computer vision research [HAN78a,b,HAN87,RIS87] it is clear that such systems are fundamentally limited by their restricted and unreliable view of the image data, and consequently their performance must suffer by the degree to which the image descriptions fail to support the system's goals.

More recently, multiple sensors have become more readily available and many algorithms have been developed for processing each type of sensor data. For example, depth maps can be obtained directly from laser range data, and indirectly from motion and stereo algorithms that are applied to pairs and sequences of images, respectively. It has also become evident that each low-level process extracts only partial descriptions of the underlying image structure, and that there is a great deal of redundancy which can be profitably exploited across these descriptions. Consequently, the need is becoming more acute for computer vision systems to fuse the information extracted by different types of low-level vision algorithms into more coherent descriptions. Maximum reliability can only be achieved through processes that can integrate information represented in widely varying forms.

To be somewhat more specific, consider the interpretation of a road scene. The formation of a 'road' hypothesis should not be based on any single type of extracted image event (e.g. regions), but rather on an aggregation of multiple types of events (e.g. lines, regions, and surfaces) that have specific relationships to each other and which contribute to the support of the road structure. For example, one might like to find a homogeneous region of an expected intensity and color, bounded by two converging straight lines, and approximately covered by a horizontal planar surface.

Of course the reader should not be misled into an oversimplified view of the problem; there are extremely difficult low-level issues to be dealt with, such as the instability of segmentation algorithms that leads to unpredictable fragmentation of lines, regions, and surfaces [BEV87,KOH83,NAG82], and inconsistencies between the elements extracted in these representations. These are problems that are implicit in the nature of the problem of integrating unreliable information and will be true

of all approaches, not just the one presented here. Our view is that to fully integrate multiple representations there will need to be complex grouping strategies that utilize the techniques presented here as part of a knowledge-directed interpretation process [BOL87,DRA87a,b,HAN87,REY87b].

In this paper, we take the view that information fusion can be accomplished during later stages of the interpretation process, rather than when the image events are first extracted (e.g. by attempting to directly integrate region and line algorithms). We also believe this will avoid some of the severe ambiguity problems encountered when performing interpretation on the independent representations prior to information fusion. Our approach to fusion will be illustrated here by extending a constraint-based object hypothesis system [RIS87] to operate over multiple token types, in this case regions and lines extracted from the image data. The construction of the region and edge representation makes use of two low-level algorithms: a local histogram-based region segmentation algorithm [BEV87,KOH83,NAG82] and a straight line extraction algorithm [BUR86]. Examples of these two processes are shown in Figure 5b,c. When surface elements extracted from depth maps are available, they can be aggregated with regions and lines. The techniques could also be easily extended to include fusion of information from textured areas, corners, volumes, and generally any other token abstracted from the same or other sensory sources.

There have been a few attempts to integrate results from multiple low-level processes operating on one or more sensory sources [HAN78b,KOH83,NAS83]. In the past, efforts to combine multiple processes operating on visual data have typically involved the integration of line and region data, which are the two most common types of low-level algorithms employed. More recently, there has been an increasing number of efforts to combine range and visual data [ARK87,BES85,SHA86]. Shafer and Thorpe [SHA86] developed a blackboard system for the CMU NAVLAB mobile vehicle; in this system, range data and visual data are independently processed and combined during the interpretation process. On the other hand, Nandhakumar and Agarwal [NAN87] combined the processing of infrared and visible light images with computational models of the image generation process to improve the results beyond that achievable by either process alone.

2 Background

2.1 The Intermediate Symbolic Representation (ISR)

The most general model of image interpretation involves the construction of a symbolic three-dimensional description of a scene from an image or set of images; a related goal is the identification of a specific 'target' object from background clutter. It is generally accepted that a computer vision system must perform a variety of transformations of the data during this interpretation process. Consequently, at a coarse conceptual level, the VISIONS image understanding system is organized into three levels of processing: low, intermediate, and high as shown in Figure 1. Currently, the low-level, or segmentation, processes output a symbolic representation of the data in the form of regions and lines. Attributes, such as color, texture, location, size, shape, and orientation, are then calculated for each region or line. Interpretation processes use knowledge of the objects in the domain to control a set of intermediate-level processes for generating initial object hypotheses and reorganizing the low-level data. World (domain) knowledge is then responsible for resolving these hypotheses data into a consistent model of the scene.

One of the key abstractions is the transformation of pixels, or more generally arrays of sensory data, into image events which can be named and accessed by their properties. We refer to the symbolic representation of an extracted image event, such as a particular region, line, curve, rectangle, or surface, etc., as a *token*; attributes are associated with tokens, and tokens participate in relationships with other tokens. Note that tokens may be defined in terms of other, more primitive tokens, as in the case of a rectangle. A *tokenset* is a collection (set) of tokens of the same type (e.g. region tokens). Tokensets, tokens, token attributes, and relations between tokens are all organized into a type of relational database called the "intermediate symbolic representation" (ISR). The ISR allows flexible associative access to tokens and serves as the communication interface between the low-level descriptive processes and the high-level interpretive processes in the VISIONS system. In general, the only requirement for placing a new type of low-level token into the ISR is that each primitive element of that data type must have a symbolic name (e.g., region-240, surface-38, corner-46) and a non-empty set of attribute-value pairs. It is the values of the token attributes, and, as we shall present here, the relational information between tokens, that provide the basis for initial interpretation processes.

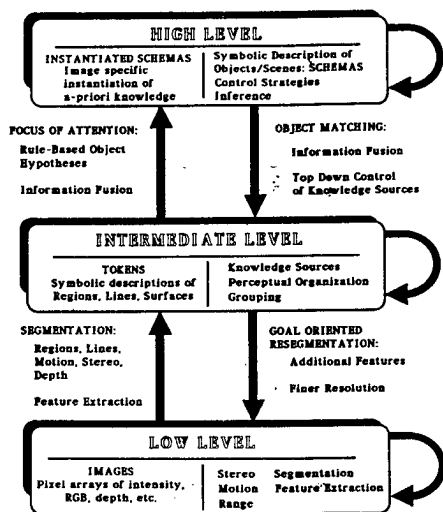


Figure 1. The VISIONS System: Processing and control across multiple levels of representation are depicted in this system overview.

3 Constraints on Tokens of a Single Type

3.1 Constraint Functions on Token Attributes

A simple type of knowledge source for generating hypotheses of object class labels for particular regions has been under development in the VISIONS environment for some time [BEL86,HAN87a] [REY87a,RIS84,WEY83,WIL81]. The general idea is to develop a mapping from a region token and its attributes onto an object label hypothesis for the region, e.g. 'grass'. We note that the VISIONS system operates primarily in the outdoor scene domain, but the general techniques developed below are applicable to most scene domains and sensor modalities. The mapping was accomplished by defining constraints on the range of an attribute from a sample set of the object and a constraint function which mapped from the region attribute into a weighted 'vote' or 'score' for the object label (see Figure 2). Compound constraints were defined as (possibly recursive) combinations of the output of a set of these 'simple' constraint functions. Region features such as region color, texture, shape, size, image location, and relative location to other objects were used. More recently, the approach has been extended to lines, using attributes such as length, orientation, contrast, width, etc. While no single constraint on the features of a region or line can ever be totally reliable, the combined evidence from many such constraints often imply the correct interpretation of a token; for example, in a rank ordering of a set of regions on the basis of the final 'score', the region-object label association for those regions near the top of the list is often correct. In many cases, it is possible to define constraints which provide evidence, in the Dempster-Shafer sense, for and against the semantically relevant concepts representing the domain knowledge [REY85,REY87a].

Rather than viewing the application of the constraint set through the constraint function as a classification process in the pattern recognition sense, the rank-ordered output can be used as an unreliable set of hypotheses and used to trigger focus of attention mechanisms in an artificial intelligence sense [HAN78b]. They are used by other more complex knowledge-based processes in a hypothesize and verify control structure [DRA87a,b,HAN87a] [WEY86].

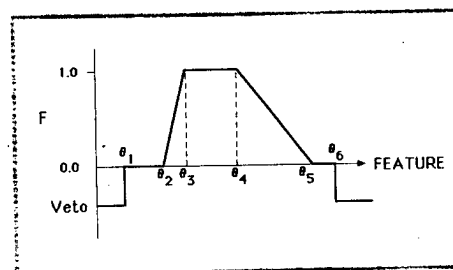


Figure 2. Structure of a simple constraint function as a piecewise linear function F mapping an image feature measurement into support for an object label hypothesis. It is specified via 6 points $\{\theta_1, i = 1, \dots, 6\}$.

A simple constraint function (which hereafter will often be referred to simply as a constraint) is a function F applied to the k^{th} attribute (or feature) of the j^{th} token of type T . Thus, if the k^{th} attribute of line tokens (type L) is length and L_{jk} is the length of the j^{th} line token, then $F(L_{jk})$ would be the response of the constraint function when applied to the line length of token j . A variety of forms for the function F have been employed, with no appreciable difference in the results. The first was an extended real-valued piecewise-linear function F

$$F(T_{jk}) \in \{[0, 1] \cup \text{VETO}\} \quad (1)$$

specified by six points in the feature range $\{\theta_i, i = 1, \dots, 6\}$ as shown in Figure 2 [RIS87, WEY83, WEY86]. The simplicity of this approach is that F in this form could be compactly stored as a 6-tuple, or sets of 6-tuples.

Compound constraints are a hierarchical collection of simple and compound constraints with an arithmetic or logical combination function for collecting the individual responses into a single response. For discussions of variations on compound constraints see [HAN87a, KIT86, RIS84, RIS87, WEY86]. In some experiments, the top-level compound constraint for an object was structured as a combination of five other compound constraints (as shown in Figure 3) to represent color, texture, size, shape, and location constraints, each of which was composed of a set of simple constraints [RIS87].

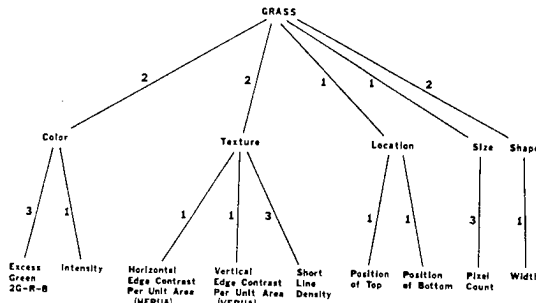


Figure 3: The structure of a compound constraint for grass showing the five component constraints defined on region attributes.

3.2 Relational Similarity Constraints on Tokens of the Same Type

When dealing with unreliable segmentation processes, forming aggregations of tokens is usually necessary since the set of tokens need to be grouped and reorganized in order to match an object model [DRA87b, HAN87a, HAN87b, REY87b]. The basis of this grouping usually involves not only the attributes of tokens, but also the relations between tokens. The constraints described in the previous section are unary, since they accept a single token attribute as input and return a value that can be viewed as a confidence or rating for the hypothesized object. The highest ranked of these hypotheses can serve as a partial (and probably errorful) interpretation of the original image. However, it is clear that constraints on relationships between tokens are also fundamental to object recognition. They can be handled in much the same way by defining constraints on relational measures (a function which quantifies a particular relation) between pairs of tokens. In this case, the response of a constraint function specifies the degree to which the constraint on a relational measure is satisfied.

Let us consider as a simple example a line token set and a relational measure defined by the simple absolute difference of their orientation attribute. Given a specific line token (e.g. L-435), all other tokens can be rank-ordered relative to L-435 by a constraint function applied to the orientation difference. Figure 4 shows two different constraints F_1 and F_2 for processing line tokens relative to a given line token. F_1 gives a maximum response of 1 for all tokens that are within 10° of L-435, a linear decreasing response from 10° to 30° and a veto response beyond 45° . The effect of applying F_1 is to rank equally all lines that are very similar in orientation to that of L-435; beyond 10° they are ordered based on their relative orientation. Lines whose relative orientation is greater than 45° away from that of L-435 are excluded. The constraint embodied in F_2 results in the selection of all line tokens that are within 5° of being orthogonal to L-435. In effect constraint F_2 defines a relation on approximately orthogonal pairs of lines that have L-435 as a member; the relation is defined to have a value of True for all pairs where $F_2(L_{435,k}; L_{j,k}) = 1$. In the

case of constraint F_1 , the relational measure for those lines not vetoed is mapped into a response which can be used to coarsely rank order the line pairs in terms of how 'parallel' they are. Note that applying a threshold to the responses produces a true relation. Either of the subsets resulting from application of these constraints could be followed by a token attribute constraint for

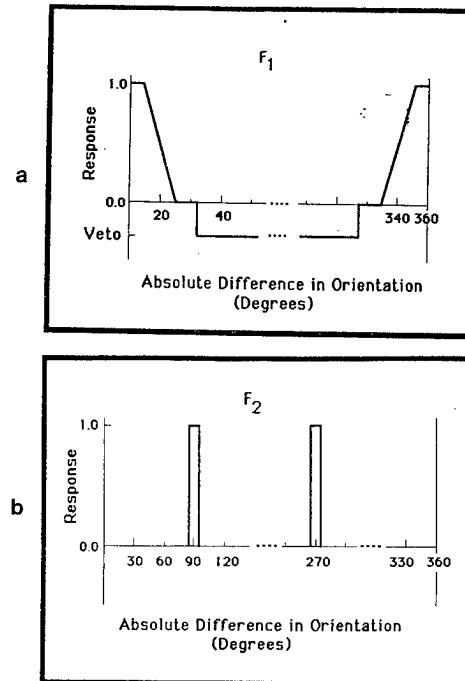


Figure 4: Example Constraint Functions on Relational Measures. The structure of two constraint functions for relating one line token to another is shown. The constraint is applied to the difference in orientation of two line tokens. (a) F_1 equally ranks all lines whose orientation is within $\pm 10^\circ$ of the given line; the response falls off as a function of the difference in orientation. (b) F_2 equally ranks all line tokens that are within $\pm 5^\circ$ of being orthogonal to the given token.

ordering or filtering the remaining lines on other attributes such as location, contrast, length, etc. in absolute terms or relative to L-435. In the next section we show how this general idea can be extended to tokens of different types, resulting in a fusion of the information from the two different sources.

4 Integrating Representations via Construction of Token Aggregates

The fundamental problem that is being addressed in this paper is the integration of multiple low-level representations into the interpretation process. While the approach presented here offers only one type of information fusion mechanism and deals with only some of the most general levels of the information fusion problem, there are several important advantages. First, it offers an entirely modular and natural method for incorporating additional processes and representations as a vision system undergoes incremental development; in particular, existing low-level representations do not have to be modified in any way. Secondly, the integration is accomplished at the intermediate grouping and/or interpretation levels through constraints which relate entities in the independent representations. Since there is no direct interaction of the processes which initially create the tokens, the mistakes of one low-level process will not affect the output

of the other low-level processes. If a sufficient body of consistent information exists in several representations, then low-level mistakes in a given representation may be detected and either ignored or corrected, as opposed to integrating partially erroneous data in some form, such as a least-squares optimization process. Third, this approach is an extension of an approach applied to token attributes that has already proven to be reasonably effective on very complex natural scenes [HAN86,HAN87a,RIS84]. Finally, the techniques can be used as part of general intermediate grouping processes. The grouping can be viewed as knowledge-directed (e.g. via a model of an object), or it could be viewed as a data-directed token aggregation process whose goal is to extract interesting structures of a priori importance [REY87b].

4.1 The Formation of Aggregations of Tokens

Relational constraints are defined as a real-valued function on a relational measure defined over a token set. A *relational measure* $M(T_{i_1 k_1} \dots T_{j_m k_m})$ is a function of the attributes of multiple tokens, possibly of different types. As we have already discussed, for tokens of the same type there is an implicit binary relational measure in that the same scalar-valued attribute of any two tokens can be compared by their similarity or difference. For example, given a specific region, similarity relational measures be used to compute the distance between feature centroids as well as the difference in mean intensities between the given region and all other regions. Once such a scalar relational measure exists, a constraint function can be applied to the relational measure to produce a response that represents the degree to which the relational constraint is satisfied.

In order to compare tokens of different types a relational measure must be defined between each pair of token types. Therefore, if new sensors are added, and new sensory events are extracted, relational measures must be defined between the existing token types and the new token types. This will allow information from new representations to be integrated. (In this paper, only binary measures between region and line tokens are developed.) The constraint functions on relational measures can then be applied to sets of tokens across the multiple representations in order to group tokens into aggregations. For example, relational measures between line tokens and region tokens can be defined, such as the degree of intersection between tokens; a constraint on this relational measure could then select, for each region, all lines that are sufficiently interior to the region. The reader should note that when a relational constraint is used to filter the token tuples, into disjoint sets, the result is a relation.

4.2 Region-Line Relational Measures

We now present a specific set of relational measures to provide a computational method of relating regions and lines. Relational constraints on these relational measures will then be used to implement the following relations between regions and lines:

- BOUNDING lines – those lines associated with a region boundary;
- INTERIOR lines – those lines interior to a region; and
- OTHER lines – those lines which intersect a region, but are neither bounding nor interior to the region.

The relational measures chosen are based on the intersection of sets of pixels. Only lines which intersect a given region are of interest. We will represent lines by the subset of pixels (called the line-support-set,) comprising that portion of the intensity surface that led to the extraction of the line [BUR86]; thus, the line-support-sets of pixels can be expected to overlap the regions that they bound or are interior to. Consequently, INTERSECTION (defined in the usual way on pixel subsets) becomes a natural relation which can be used as a filtering constraint to select a subset of tokens. In the following discussion, however, three other relational measures will be defined and used to implement

the three relations mentioned above: "interior-line-percentage", "region-perimeter-percentage", and "line-boundary-percentage".

The first relational measure, "interior-line-percentage", is the ratio of line area interior to the region to total line area. The interior-line-percentage measure discriminates lines that are entirely INTERIOR from BOUNDING lines, whose line-support set will lie partially outside the region. An INTERIOR line will have a value of 100% for this relational measure, indicating that the line-support-set is completely contained by the region. An ideal BOUNDING line with a symmetric line-support-set of pixels lying exactly on the region boundary would have half its pixels in the region and a value of 50% for its interior-line-percentage.

The other two relational measures can be used to discriminate BOUNDING lines from INTERIOR lines. The natural duality between regions and their boundary lines can be exploited in a straightforward manner to indicate how much of a region boundary or a line is covered by the other. "Region-perimeter-percentage" measures the fraction of a region boundary made up of one line and is defined to be the ratio of the intersection of the region perimeter pixels and the line-support-set to the length of the region perimeter. "Line-boundary-percentage" measures the fraction of a line contributing to the region boundary and is defined to be the ratio of the intersection of the line-support-set and the region perimeter to the total line length in pixels. Ideally, a line which lies approximately on a region boundary will have a high value of line-length-percentage since the region boundary will cover much of the line. The same is true of region-perimeter-percentage although a single line will be expected to cover a smaller portion of the entire region boundary.

4.3 Relational Constraints

Relational constraints are used as the final step in the formation of aggregations from multiple representations. The relational measures presented in the preceding sections provide the basis for defining these relational constraints.

A relational constraint function for lines and regions can be specified for each relational scalar measure that has been defined (in the same manner that token attribute constraints are defined). Thus, a simple constraint can be specified for each of interior-line-percentage, line-length-percentage, and region-perimeter-percentage measures; note that any of these simple constraints may be omitted. A combination function can then be defined for combining the output of the set of simple constraints into a compound relational constraint.

The form of the function that combines the simple constraints is not critical, and in this paper we will use the same simple piecewise-linear function described earlier with a range of $\{[0,1] \cup \text{VETO}\}$. The VETO range(s) serves only as a first filter for selecting or removing candidates for processing, in the sense that the vetoed tokens do not satisfy the constraint. This does in fact, define a relation over the token sets, but the remaining non-vetoed tokens in the relation still have the graded response from the constraint function, which can be used for ranking or further filtering of token pairs to produce a more restricted subset of tokens.

4.4 Controlling the Formation of Token Aggregations

The aggregation of tokens via relational constraints must, of course, contend with the combinatorics of the large number of image tokens whose relationships must be examined. The concept of *focus-of-attention* becomes important when one considers that the representations being used typically involve 2,000 to 10,000 lines and 200 to 1,000 regions. Thus, there are potentially 400,000 to 10 million line-region pairs which could be related, and these numbers become much larger as additional representations, multiple sensory sources, finer image resolutions, or larger images are considered.

The order and manner in which attribute constraints and relational constraints are applied are the basis of the *control strategy* for the construction of token aggregations. Tokens of different types can be aggregated in many ways and one seeks to avoid

the combinatorics of computing relations between large token sets. There are two obvious ways in which the constraint functions can be used in control. A constraint on the cross product of the token sets can be used either to *rank order* the set of token tuples via the response of the constraint function, or to *filter* (i.e. select) a subset of the token tuples for further processing. Of course the responses of the constraint functions that are used for ordering could also be used for filtering by specifying filtering criteria such as thresholds. Example control strategies for limiting the computation are outlined in [RIS87].

5 Results

This section presents the results of forming aggregations via token attribute and token relational constraints applied to suburban house scenes and road scenes; Figure 5 shows a representative image of one of the house scenes and a typical region segmentation and line description for the image. The examples chosen for this paper involve aggregations of tokens that serve as texture measures and aggregations of tokens with specific shape properties.

A simple texture measure based on line density can be computed by counting the number of lines within a region and normalizing by the region size. This is accomplished by forming

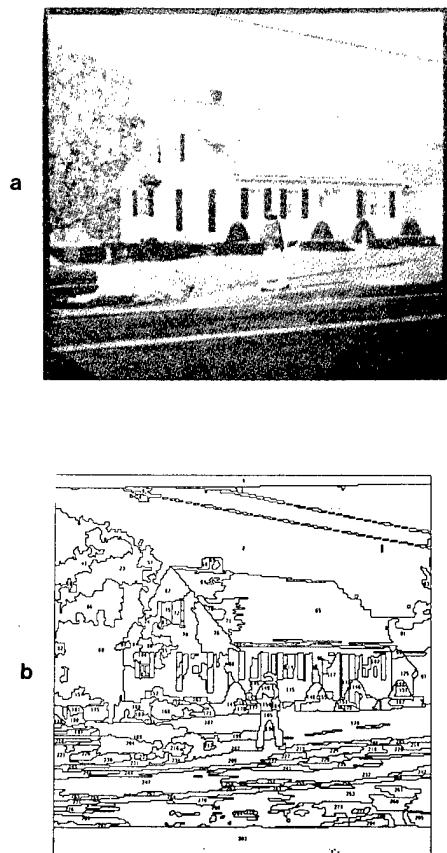


Figure 5. Results. (a) a black and white rendering of the original color image; (b) regions produced by a segmentation system using localized histograms followed by region merging; (c) straight lines produced by an algorithm which uses similarity of gradient orientation as the primary organizational feature.

aggregations of regions and their interior lines. A filtering constraint uses the interior-line-percentage relational measure to select only those lines which are completely (or mostly) interior to the region. A filter is defined to group into an aggregation those lines associated with each region which sufficiently satisfy both relational and attribute constraints. The density of interior lines in each aggregation is then computed as an attribute of this new token and mapped to a score for the region (which can also be thought of as a score for the region-line aggregation).

Figure 6a shows an example of extracting interior lines for regions in a house scene, and then computing the interior line density of these regions as a texture measure (see Figure 6b). Some objects, notably the roof of a house, are characterized by short horizontal lines (due to the shingles) interior to the region. By adding the additional attribute constraint of horizontal orientation on the interior lines, the previous result can be extended to focus attention on the house roof as shown in Figure 6c,d. Additional constraints on line length and line contrast can be defined to extract only short, horizontal interior lines to the degree that these characteristics of the expected texture element are known.

The roof region could be obtained or verified in another way. The line-boundary-percentage relational measure could be used to select lines which lie to a great extent on the boundary of the region (see Figure 7a). A line attribute rule could then be defined to favor long lines (see Figure 7b). The lines which received high scores from both the line boundary rule and the line length rule (i.e., long boundary lines) could then be grouped to form a region-boundary aggregation. At that point parallel relations, rectangle or parallelogram structures could be identified.

Another simple shape measure can be computed by determining if a region is bounded by a pair of long vertical lines. As Figure 8 shows, the process is useful for extracting telephone poles in road scenes. The filtering relational constraint for this measure uses line-boundary-percentage to select only those lines which lie on a region boundary; an attribute constraint then selects the long vertical lines. Relational measures can be defined to form aggregations of pairs of parallel overlapping lines from the long vertical boundary lines.

A variety of more complex 2D shapes can be matched to lines by extracting lines that bound regions. The techniques presented here are only a part of more complex grouping and model matching procedures that are being developed in other research [HAN87a,REY87].

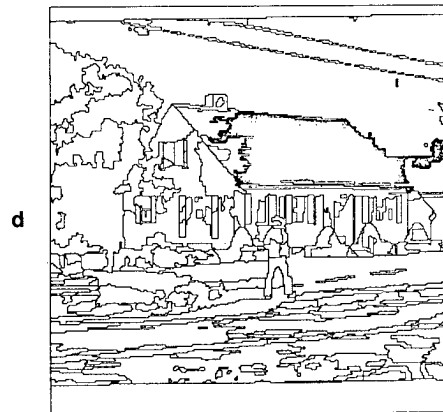
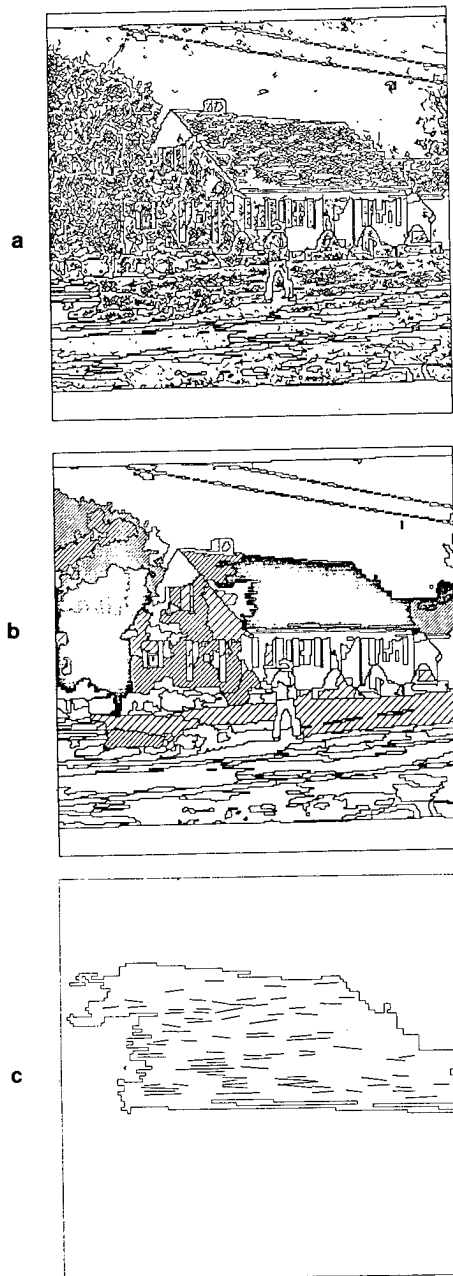


Figure 6. Example of Texture Measure for Extracting a House Roof. A simple texture measure computed by a relational constraint based upon the density of lines within a region. (a) Lines which received a high score on the relational measure of interior-line-percentage (i.e. INTERIOR lines); (b) The density of interior lines for each region represented by the density of shading; (c) Horizontal INTERIOR lines for the roof region; (d) Density of horizontal interior lines.

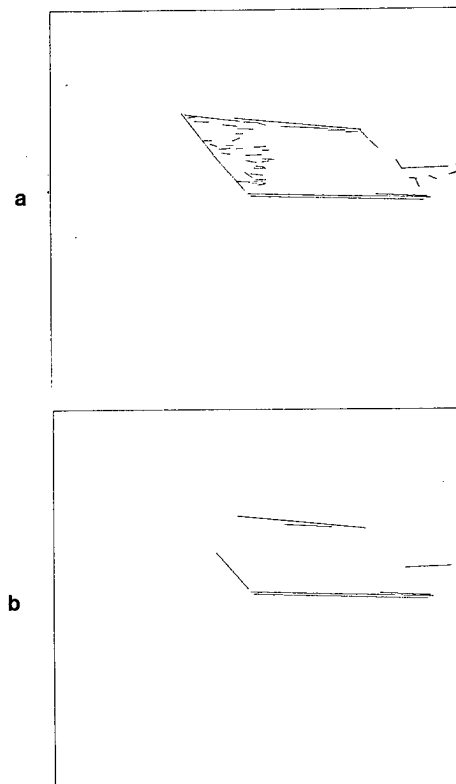


Figure 7. Extracting Roof via Long Bounding Lines. Given a possible roof region the long bounding lines can be filtered to find a roof shape. (a) Lines bounding the hypothesized roof region, and (b) Long bounding lines which can be the basis of forming an approximate parallelogram shape.

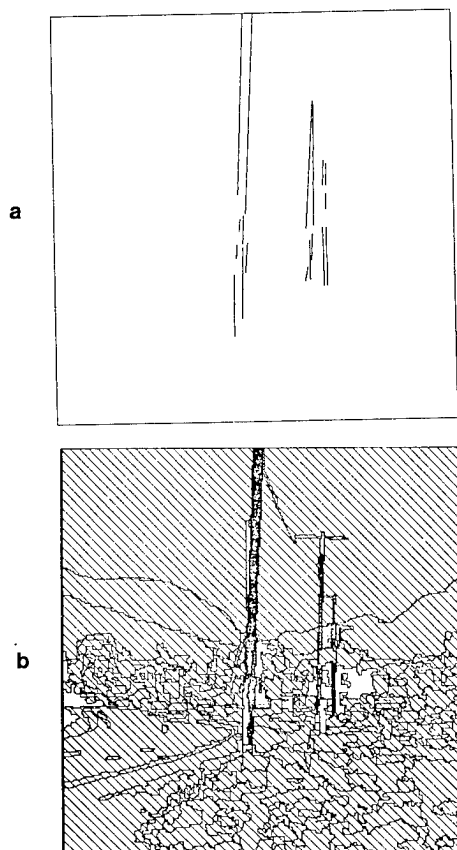


Figure 8. Example of Extracting Telephone Poles via Vertical Bounding Lines. (a) The line pairs formed by the line constraint of bounding vertical lines. (b) The relational measures are mapped back to the regions; hatched regions have no long vertical bounding lines and are vetoed.

Figure 9 shows a set of rectangles extracted from a house scene. In this case, the set of lines intersecting a region were filtered to extract the set of bounding lines. These were further filtered on the basis of co-parallel, collinear, and endpoint coincidence relations. Pairs of 'adjacent' lines were then filtered on the basis of constraints on their relative orientation in order to form corner hypotheses and the resulting set of lines were matched to a rectangle model. As the figure indicates partial matches to the rectangle model are allowed. The matches could be further restricted if one is seeking dark shutters by using a constraint on the intensity of regions, although this was not done in this example.

There are sometimes serious problems with constructing aggregations through relational measure directly computed from initial token representations. If the desired primary token is fragmented, whether it be a region or a line, then the expected relational responses might be distorted significantly because some of the expected token attributes, token relations, and features of the extracted aggregation may be significantly changed. One must balance the unreliability of extracting useful primary tokens by the computational savings achieved by focussing upon the subset of secondary tokens that satisfies some relational constraints with respect to the filtered primary set. To the degree that these problems occur, many stages of hierarchical token aggregation may be necessary, perhaps using more complex strategies for applying relational measures and grouping tokens.

Let us consider a specific real example, shown in Figure 10, where both representations (e.g. regions and lines) would have some difficulty in directly providing the basis for aggregations of

tokens from the other representations. The roof region in Figure 5b is fragmented into many smaller regions (Figure 10a). In this case no region will get full benefit of the bounding roof lines, since each region is only bordered by a subset of the roof lines; Figure 10b depicts the lines intersecting the largest roof region. An alternate grouping strategy would be to extract a set of roof lines first (which may be a difficult task in itself), then the set of regions that are bounded by these lines could be used in some manner to aggregate regions. The line information if properly filtered shows the outline of the roof fairly clearly (Figure 10c), but the initial set of lines in that spatial vicinity provide many possible aggregations. The process of grouping lines into meaningful geometric structures is a non-trivial problem and is a focus of continuing work on grouping and knowledge-directed processing in our research group [BOL87,HAN87a,HAN87b,REY87,WEI86,WEY86].

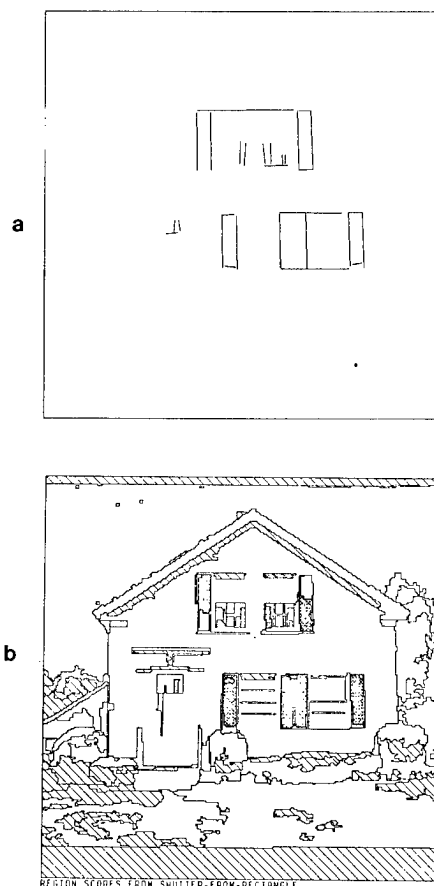


Figure 9. Extracting Rectangular Window and Shutter Hypotheses. (a) Horizontal and vertical bounding lines; (b) The regions produced by a relational constraint on the extracted bounding lines.

6 Conclusion

The use of constraints on relational measures between tokens of the same and different types is a uniform, straightforward way of combining information from multiple low-level processes. The techniques developed in this paper allow information fusion to take place during the interpretation process as intermediate tokens are aggregated via object-dependent constraints on token attributes and token relations. The ideas presented here can be

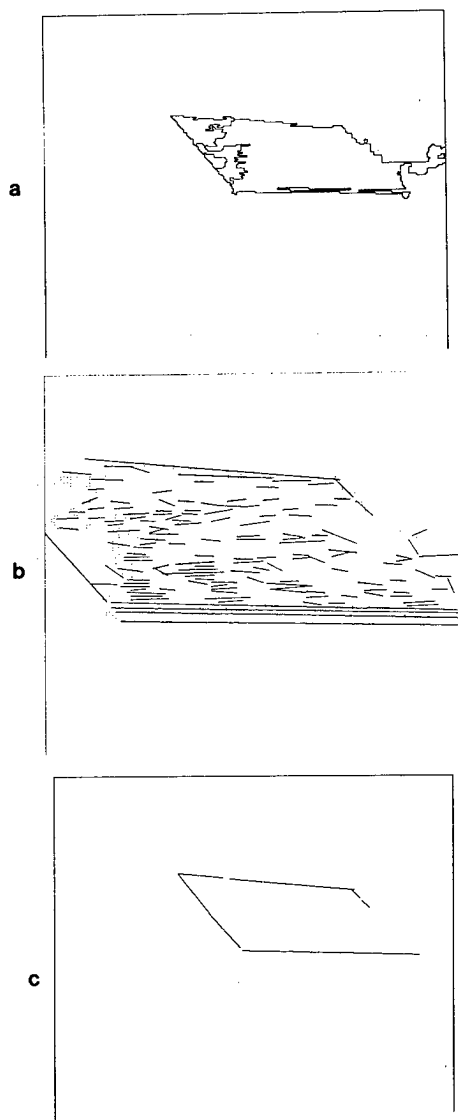


Figure 10. Example of difficulties in aggregating tokens. (a) The roof is fragmented in the region representation; (b) The lines intersecting the largest roof region only capture a portion of the relevant line information; note the two key missing lines at the upper left corner of the roof. (c) A subset of the full set of lines which, if they could somehow be selected, would provide the appropriate line aggregation to group the roof regions, and allow the roof outline to be completed in a straightforward manner [WEY86].

used to build hierarchical aggregations; for example, aggregates of lines could be formed by grouping colinear sets of lines into new longer line tokens [e.g., BOL87, REY87b, WEI86]. By treating each aggregation as a new token, attributes could then be computed for each and the constraints applied recursively.

The types of information that could be added include surface segmentations, and 2D and 3D motion and depth token attributes. Each segmentation or low-level process would create a set of tokens with associated attributes which could be added to the intermediate-level representation. These new tokens could then be used in the same way as regions and lines are used now.

Each new token type would require the definition of relational measures between tokens of different types. There would be no other major modifications to the system. The approach overlaps issues and techniques in the areas of grouping and model matching. To the degree that tokens in one of the several representations do not exhibit the characteristics that provide the basis for directly extracting the desired structures, more complex perceptual organizing processes and knowledge-based strategies will be required. Each object that must be recognized could be defined by a separate model and control strategy for aggregating the different token types. While the concepts of relational measures and relational constraints can still be the basis of these strategies, many stages of hierarchical aggregations may be required. In such cases efficient control strategies will become a major issue. Thus, multiple alternate grouping strategies will probably be required in order to extract and utilize information across multiple representations in a generally robust and efficient manner. In the VISIONS system [DRA87a,b, HAN87a], the knowledge-based schema system provides flexible mechanisms for defining and applying control strategies. The mechanisms described in this paper are only meant to serve as the first stage of this organizing process.

7 Acknowledgements

The authors would like to express their appreciation to Amerinex Artificial Intelligence Corporation (AAI) for KBVision, a commercially available, integrated, graphically-oriented environment supporting basic research at all three levels of the image understanding hierarchy and to Rob Belknap (AAI) for many of the results presented in the paper. They would also like to thank Laurie Waskiewicz at the University of Massachusetts for her efforts in preparing this manuscript.

References

- [ARK87] R. Arkin, "Towards Cosmopolitan Robots: Intelligent Navigation in extended Man-Made Environments", Ph.D. dissertation, Computer and Information Science Department, University of Massachusetts at Amherst, expected in August 1987.
- [BEL86] R. L. Belknap, E. M. Riseman, and A. R. Hanson, "The Information Fusion Problem and Rule-Based Hypotheses Applied to Complex Aggregates of Image Events," *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, June 1986.
- [BES85] P.J. Besl, and R.C. Jain, "Range Image Understanding", in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA, June 9-13, 1985, New York, pp. 430-451.
- [BEV87] R. Beveridge, A. Hanson, and E. Riseman, "Segmenting Images Using Localized Histograms and Region Merging", submitted to *International Journal on Computer Vision*, Spring/Summer 1988, also COINS Technical Report 87-88, Computer & Information Science Department, University of Massachusetts at Amherst.
- [BOL87] M. Boldt and R. Weiss, "Token-Based Image Abstraction", COINS Technical Report, Computer & Information Science Department, University of Massachusetts at Amherst, July 1987.
- [BUR86] J. B. Burns, A. R. Hanson, and E. M. Riseman, "Extracting Straight Lines," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. PAMI-8, pp. 425-455, July 1986.

- [DRA87a] B.A. Draper, R.T. Collins, J. Brolio, J. Griffith, A. Hanson, and E. Riseman, "Tools and Experiments in the Knowledge-Directed Interpretation of Road Scenes", *Proc. of the DARPA Image Understanding Workshop*, Los Angeles, CA, February 1987, pp. 178-193. Also COINS Technical Report 87-05, Computer & Information Science Department, University of Massachusetts at Amherst, January 1987.
- [DRA87b] B.A. Draper, J. Brolio, R. Collins, A. Hanson, and E. Riseman, "The Schema System", submitted to *International Journal on Computer Vision*, Spring/Summer 1988, also COINS Technical Report, Computer & Information Science Department, University of Massachusetts at Amherst, in preparation.
- [HAN78a] A. R. Hanson and E. M. Riseman, "VISIONS: A Computer System for Interpreting Scenes," *Computer Vision Systems* (A. Hanson and E. Riseman, eds.) (1978), pp. 303-333, Academic Press.
- [HAN78b] A. R. Hanson and E. M. Riseman (Eds.), *Computer Vision Systems*, New York, Academic Press, 1978.
- [HAN87a] A. R. Hanson and E. M. Riseman, "The VISIONS Image Understanding System - 1986", in *Advances in Computer Vision*, (Chris Brown, Ed.), Erlbaum Press, also COINS Technical Report 86-62, Computer & Information Science Department, University of Massachusetts at Amherst, December 1986.
- [HAN87b] A.R. Hanson and E.M. Riseman, "From Image Measurements to Object Hypotheses", COINS Technical Report 87-129, University of Massachusetts at Amherst.
- [KIT86] L. Kitchen, R. Weiss and J. Tuttle, "Identification of Human Faces Using Data-Driven Segmentation, Rule-Based hypothesis Formation, and Interactive Model-Based Hypothesis Verification", COINS Technical Report 86-53, Computer & Information Science Department, University of Massachusetts at Amherst, October 1986.
- [KOH83] R. R. Kohler, "Integrating Non-Semantic Knowledge into Image Segmentation Processes," Ph.D. Thesis, University of Massachusetts at Amherst, September 1983; also COINS Technical Report 84-04.
- [MCK84] D. M. McKeown, W. A. Harvey and J. McDermott, "Rule Based Interpretation of Aerial Imagery," Dept. of Computer Science, Carnegie-Mellon University, (September 1984).
- [MCK85] D. M. McKeown, W.A. Harvey and J. McDermott, "Rule Based Interpretation of Aerial Imagery", *IEEE PAMI*, Vol. PAMI-7, No. 5, September 1985, pp. 570-585.
- [MCK86] D. M. McKeown, C.A. McVay and B.D. Lucas, "Stereo Verification in Aerial Image Analysis", *Optical Engineering*, Vol. 25, No. 3, March 1986, pp. 333-346. Also available as Technical Report CMU-CS-85-139.
- [NAG82] P. A. Nagin, A. R. Hanson, and E. M. Riseman, "Studies in Global and Local Histogram-Guided Relaxation Algorithms," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 3 (May 1982), pp. 263-277.
- [NAN87] N. Nandhakumar and J.K. Aggarwal, "Multisensor Integration-Experiments in Integrating Thermal and Visual Sensors", *Proc. IEEE ICCV*, London, England, June 8-11, 1987, pp. 83-92.
- [REY87a] G. Reynolds, N. Lehrer and J. Griffith, "A Method for Initial Hypothesis Formation in Image Understanding", *Proc. of the DARPA Image Understanding Workshop*, Los Angeles, CA, February 1987. Also COINS Technical Report 87-04, Computer & Information Science Department, University of Massachusetts at Amherst, January 1987.
- [REY87b] G. Reynolds and J. Ross Beveridge "Geometric Line Organization Using Spatial Relations and A Connected Components Algorithm", COINS Technical Report, Computer & Information Science Department, University of Massachusetts at Amherst, in preparation, 1987.
- [RIS84] E. M. Riseman and A. R. Hanson, "A Methodology for the Development of General Knowledge-Based Vision Systems," *IEEE Proc. of the Workshop on Computer Vision: Representation and Control* (1984), pp. 159-170.
- [RIS87] E. M. Riseman and A. R. Hanson, "A Methodology for the Development of General Knowledge-Based Vision Systems", in *Visions, Brain, and Cooperative Computation*, (M. Arbib and A. Hanson, Eds), MIT Press, 1987, pp. 285-328. Also COINS Technical Report 86-27, University of Massachusetts at Amherst, July 1986.
- [SHA86] S.A. Shafer, A. Stentz, and C.E. Thorpe, "An Architecture for Sensor Fusion in a Mobile Robot", *International Conference on Robotics and Automation*, San Francisco, CA, 1986, pp. 2202-2011.
- [WEI86] R. Weiss and M. Boldt, "Geometric Grouping Applied to Straight Lines", *Proc. of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Miami, FL, June 22-26, 1986, pp. 489-495.
- [WEY83] T. Weymouth, J. Griffith, A. R. Hanson, and E. M. Riseman, "Rule Based Strategies for Image Interpretation," *Proc. AAAI-83* (August 1983).
- [WEY86] T. E. Weymouth, "Using Object Descriptions in a Schema Network for Machine Vision," Ph.D. Thesis, University of Massachusetts at Amherst (April 1986). Also COINS Technical Report 86-24, Computer & Information Science Department, University of Massachusetts at Amherst, May 1986.
- [WIL81] T. Williams, "Computer Interpretation of a Dynamic Image from a Moving Vehicle," Ph.D. Thesis and COINS Technical Report 81-22, University of Massachusetts at Amherst (May 1981).

**DYNAMIC EVALUATION OF INFORMATION SOURCES
IN RULE-BASED SENSOR FUSION SYSTEMS**

H. James Antonisse
Karl S. Keller

The MITRE Corporation
7525 Colshire Drive
McLean, VA, 22102
(703) 883-7887/883-7626

Abstract

Knowledge-based programming techniques are playing an increasingly significant role in computer applications. Recently a number of knowledge-based approaches to integration of multiple sensor sources for tactical intelligence have been explored. However, several factors threaten to inhibit more widespread applications of knowledge-based technology in this area. One of these is the difficulty of validating the enormous amount of knowledge needed for operational systems. We discuss an approach to dynamically measuring the quality of knowledge in systems that express their knowledge in the form of if-then rules. We illustrate the capability of the method for capturing dynamic system behavior, and discuss how the method can be extended to work in environments where there is no access to ground truth. Finally we discuss an extension which permits adaptive assessment of the information sources of rule-based sensor fusion systems.

DYNAMIC EVALUATION OF INFORMATION SOURCES IN RULE-BASED SENSOR FUSION SYSTEMS

H. James Antonisse
Karl S. Keller

The MITRE Corporation
7525 Colshire Drive
McLean, VA, 22102
(703) 883-7887/883-7626

1.0 INTRODUCTION

Artificial Intelligence (AI) techniques are playing an increasingly significant role in computer applications. Foremost among the methods of AI that are finding wide-spread use are those that underlie knowledge-based systems. These are programs designed so that the knowledge needed to solve problems can be expressed separately and manipulated independently of the part of the program that applies the knowledge. This separation of the knowledge from the mechanisms that apply it has proven to be an exceptionally powerful programming technique. As a result, knowledge-based technology will be an increasingly important tool in the application of computing machinery to complex and heretofore unmanageable problems.

A number of systems currently under exploratory development take knowledge-based approaches to the task of data fusion in the tactical intelligence domain. Included among the activities addressed by these systems are:

1. to correlate raw intelligence reports to lower echelon units (refs. 1, 2, and 3);
2. to aggregate lower level units into structured, higher-level echelon units (refs. 4 and 5);
3. to predict activities and motions for military units (refs. 6 and 7);
4. to identify strategies, objectives, and decision points for (usually higher-level) units (refs. 8, 9, and 10)

These developments are expected to bear fruit in the reasonably near future. They will be the underpinnings of decision aids for intelligence staffs which will help construct and evaluate hypotheses about the enemy's dispositions and probable courses of action.

Before such systems can be fielded, however, a problem of particular importance to the intelligence community must be addressed: the knowledge domain is extremely volatile. This point strongly impacts the design issues of a fully adequate decision aid system for intelligence staff. Not only will the crucial knowledge components of such systems refer to uncertain elements of the enemy's behavior such as doctrine and tendencies, but the nature of those elements is that they are constantly changing as the enemy gains experience and adopts new tactics and strategies. These factors make it extremely important to explore methods by which knowledge may be dynamically evaluated for currency, and system's behavior adapted in the face of knowledge failures.

Machine learning is the branch of artificial intelligence that encompasses adaptive behavior of knowledge-based systems. It is defined as the automated modification of a system's behavior to improve its performance. For any realistic system, the main obstacles to machine learning are the problems of (1) effectively evaluating the performance of those components of a system open to modification, and (2) finding modifications of those components which will improve performance. Thus for rule-based systems the

measure of system performance should apply to individual rules in the system since it is modifications to these that give rise to modifications of system behavior.

This paper presents research in progress in machine learning as applied to sensor data fusion. In it we discuss a dynamic approach to the measurement rule performance in data fusion systems for tactical intelligence. We then extend this approach to the problem of assessing the performance of the system's information sources. This capability is expected to be of use in two areas. It will increase the capacity of automated information integration systems to deliver high-quality support to intelligence staff and it can serve as input to collection management staff in the

The paper is organized into six sections. In Section 2.0 we briefly describe the operation of rule-based systems. We illustrate the ability of an existing algorithm, the bucket brigade, to lead to a solution of the credit assignment problem in Section 3.0. In Section 4.0 the method is extended to environments in which there is no access to ground truth. We discuss extensions which make similar evaluation possible on sources of information in Section 5.0. Section 6.0

2.0 RULE-BASED SYSTEMS

The most common method of representing knowledge inside a computer is as a set of rules. The rules are "if-then" statements describing the conditions (the "if" part, or *antecedent*, of the rule) under which the rule could be applied, and the action (the "then" part, or *consequent* of the rule) to be taken when the rule is applied. For example, a knowledge base designed to detect artillery and armored units from sensor reports might consist of the following rules:

- RULE1:** IF there is a report of a shooter at a time and location and there is a report of a mover at the same time and location
- THEN
there is a tank at the time and location
- RULE2:** IF there is a report of a shooter at a time and location
- THEN
there is artillery at the time and location
- RULE3:** IF there is a communications report at a time and location
- THEN
there is a radio at the time and location
- RULE4:** IF there is a tank at a time and location and there is a radio at the same time and location
- THEN
there is an armored unit at the time and location

RULE5: IF there is artillery at a time and location and there is a radio at the same time and location

THEN
there is an artillery unit at the time and location

These rules are over-simplified examples of the rules that would appear in an operational system. They are intended for pedagogical purposes only and in this case model an environment in which "armored units" move, shoot, and communicate all the time, while "artillery units" just shoot and communicate.

For tasks of even moderate complexity, many rules may be candidates for application at a given time. Notice, for instance, that Rule2 will have its antecedent matched any time Rule1's match. An important aspect of the intelligent behavior of a system involves the selection of the rule from among those matched that is to be fired. The firing of a rule may in turn change the set of rules whose antecedents match. The following example will illustrate the process. Suppose we receive

(Shooter reported at Time1 and Location1)

(Mover reported at Time1 and Location1)
(Communication reported at Time1 and Location1)

Rules 1, 2, and 3 have their antecedents matched by the reports. By the conventional account of conflict resolution for rule-based systems (ref. 11) the rule with the most antecedents, Rule1 in this case, is chosen for firing. In firing, it uses the shooter and mover reports to establish the existence of a tank at Location1, Time1. Rule2 is removed from consideration since the shooter report that originally made its antecedent match was used by Rule1. Rule3 still matches and it is fired, using the communications report to establish a radio at Location1, Time1. The system now holds that:

(There is a tank at Time1 and Location1)
(There is a radio at Time1 and Location1)

Rule4 now matches and is fired. It establishes:

(There is an armored unit at Time1 and Location1)

The inference process described is typical of contemporary knowledge-based systems. The main difference of actual systems and the process described above is in the amount and the types of knowledge brought to bear on a given problem. For instance, actual systems are likely to reason from a set of plausible missions the enemy may be conducting as well as positions of equipments and records of activities.

3.0 CREDIT ASSIGNMENT

One of the ways in which the knowledge-based system described above is typical is that it depends on the knowledge base being static. Once it is written it will follow the same inference paths given similar information. It has no way to modify its behavior. If it turned out, for instance, that there were no armored unit at Location1 at Time1, it would take no cognizance of its error. As mentioned above, however, the tactical intelligence domain is highly volatile in terms of the information and knowledge of the enemy. In this section we describe an elaboration of the inference procedure which evaluates the knowledge in the system

The major difficulty of effectively evaluating rules in a system is the proper apportionment of credit for successful rule actions (or assignment of blame for erroneous ones) (refs. 12-13). The problem is that any of the rules leading to the action may merit the most responsibility or culpability for the action. Consider the example above. If no armored unit is subsequently found at the specified time and location, any one of Rule1, Rule2, Rule4, or any of their combinations might be to blame for the error. For complex knowledge systems, this becomes an extremely difficult problem.

3.1 The Bucket Brigade Algorithm

We have been exploring an approach to this problem based on the bucket brigade algorithm (refs. 14-15). The bucket brigade was developed originally in the context of work on population genetics to simulate the fitness of individuals in a population to reproduce (ref. 16). In our work, it functions by simulating an artificial economy among the rules in the system. Through a competitive, bidding-and-payoff process, the rules accumulate or lose capital based upon the contribution to useful system behavior. The strength of a rule as measured by its capital, called its *wealth*, is thus a measure of its utility. In the classic bucket brigade, utility derives ultimately from a scheme of rewards administered to the system for "good" actions. These rewards are called "environmental payoffs" because they derive from the direct interaction of the system with the environment. The basic process is explained below, while Section 4.0 contains a description of an "unsupervised" implementation of the algorithm that does not depend on direct intervention ("supervision")

We illustrate the bucket brigade using our previous example. Each rule is given an initial amount of capital. We distribute the capital carefully in this example so we may demonstrate interesting features of the algorithm. However, it will be seen that these initial wealths are dynamically adjusted; their initial distribution doesn't matter in the long run. Let the

RULE1:	8
RULE2:	7
RULE3:	7
RULE4:	16
RULE5:	16

intent is to model an artillery unit at Location1 plus a bogus mover report from the same location. Let the new batch of reports come in looking as follows:

(Shooter reported at Time2 and Location2)
 (Mover reported at Time2 and Location2)
 (Communication reported at Time2 and Location2)

When these are placed in working memory, Rules 1, 2, and 3 again have their antecedents matched, resulting in the following bids:

RULE1: 4
 RULE2: 3.5
 RULE3: 3.75

Again, Rule1 is chosen to fire from among its conflict set, followed by Rule2. They establish the conditions for Rule4 to fire, but because its wealth is now only 8, its bid is down to 4. It wins the bid, again it has no competitors, and pays out 2 each to Rules 1 and 3. The *wealths* of the rules in the rule base are now:

RULE1: 6
 RULE2: 7
 RULE3: 5.75
 RULE4: 4
 RULE5: 16

Rule1's wealth has started to decrease. After the second batch of reports, the failure of the inference chain to establish correct results affects the wealth of the contributing rules one step back in the inference chain. This in turn affects the ability of those rules to compete in the bidding process. Observe: when the next batch of reports comes in (identical with the first two, but for the time being Time3), the bids of Rules 1, 2, and 3 are respectively 3, 3.5, and 2.875. This time Rule2 wins the bid, establishing artillery at Location1, Time3. Rule1 is taken out of the conflict set, and Rule3 is fired. When that happens, Rule5's antecedents are matched. It bids 8, is fired, and establishes an artillery unit at Location1 Time3. This is what we have been waiting for, and we reward Rule5 with some arbitrary sum related to the importance of the result. Here, Rule5 receives an "environmental payoff" of 16.

The effects of the incorrect action have now been overcome by the work of the competitive algorithm. Continued reports will strengthen the inference chains that led to "artillery unit" until a stable set of wealths is achieved at each rule.

4.0 UNSUPERVISED LEARNING

The algorithm as presented so far is a supervised algorithm. It works only because an omniscient researcher distributes environmental payoffs on favored actions by the system. This is unacceptable for our application of interest since no access to absolute knowledge is available. Even a direct report on a state

The algorithm works by modifying the strategy by which the system decides which of a set of matching rules to fire. The set of rules matching at a given time is called the *conflict set*, and the method for choosing among them is called the *conflict-resolution strategy*. The bucket brigade implements an "economic" competition based on bidding behavior by the rules. The new view of conflict resolution is that each rule matched is competing for the privilege to establish its consequent. The bid is based on how well the rule matched the environment (its *specificity*), the wealth of the rule, and a factor c , where $0 < c \leq 1$, which restricts the percentage of its wealth a rule may risk in any one bidding cycle. Rules all use the following function to

$$\text{Bid} = (\text{specificity}) \times (c) \times (\text{wealth})$$

The specificity is 1.0 if the environment matches the rule's left-hand side exactly (versus partial or more general matches). The bid is a "closed" bid; no rule bases its bid on the bidding behavior of other rules. The highest bid in the conflict set wins. For our purposes, $c = 0.5$ and specificity = 1.0, thus the competition depends strictly on relative rule wealths. When the three reports come into the (modified) system, Rules 1, 2, and 3 make bids respectively as follows:

RULE1: 4
 RULE2: 3.5
 RULE3: 3.5

As before, Rule1 is chosen to fire from among its conflict set competitors. This time, however, some new events take place. Rule1 must pay out, from its capital, the amount of the bid. After Rule1 fires, its wealth is decremented to 4; it is at risk by the amount of its bid. Also, the result of Rule1's firing is tagged with the fact that Rule1 established it. Rule2 is taken out of the conflict set, leaving only Rule3. When it fires, its

When Rule4's antecedents are matched it enters the conflict set and makes a bid of 8. It wins the bid (by virtue of being in the conflict set alone) and must pay out its bid to establish its result -- an armored unit at Location1 at Time1. The crux of the bucket brigade method for credit assignment is that the bid that Rule4 pays out is paid to Rules 1 and 2, so their respective wealths are incremented to 8 and 7.5. Rules 1 and 2 have recouped their temporary losses; their investment has

Had there been an armored unit at the specified location, the bucket brigade, to work correctly, would require that a reward be given to Rule4 for posting a correct result. In this case, however, we are postulating that there is no armored unit at the specified location, so no reward is given to Rule4, and it loses the capital it spent on the bidding process. It is the first to feel the effects of error.

3.2 Adaptive Behavior by Competitive Conflict RESOLUTION

The fact that there was no armored unit at Location1 is reflected in the new set of reports. The

of affairs is qualified by an assessment of the position of the source to know the state of affairs, the source's competence, and the source's reliability. In this section we extend the system to operate in unsupervised environments.

4.1 Predictive Inference

In order to achieve unsupervised learning, the learning system must adjust its behavior strictly on the basis of its "sensory" input. In our example, this is the report stream represented by the mover, shooter, and radio reports. We impose the condition upon the system that a report (either predictive or retrodictive) must be issued with the establishment of every abstract concept (i.e., a concept introduced by inference from reports, such as "radio"). That is, all concepts in the system must have observational consequences at a time other than that at which it is must be established without its being of reports at some time other than that at which it is "observed".

We implement this principle through the following rules:

- RULE6:** IF there is a tank at a time and location
 THEN
 there will be a report of a shooter and a report of a mover at that time and location
- RULE7:** IF there is artillery at a time and location
 THEN
 there will be a report of a shooter at that time and location
- RULE8:** IF there is a radio at a time and location
 THEN
 there will be a communications report at that time and location
- RULE9:** IF there is an armored unit at a time and location
 THEN
 there will be a tank and a radio at the following time in a new location
- RULE10:** IF there is an artillery unit at a time and location
 THEN
 there will be artillery and a radio at that time and location

These rules are more or less the "inverse" of Rules1-5, above, taking the conceptual entities established by them back down to a predicted set of sensor reports. (The "following time and new location" referred to in Rule9 is just shorthand for the projected times and places that would be computed in a real system). The new rules need not have been such direct inverses. As mentioned before, in actual systems the analysis of expectations is expected to hinge on an analysis of the goals and the plans to achieve those goals of the objects of interest (e.g., armored-unit's missions).

The operation of this expanded rule set is just as before, except the chaining continues through the newly added rules and results in a set of propositions about reports at other times. The appropriate place for the evaluation of the system's performance has been temporally deferred through these new rules to the next time step and logically referred to assertions about the report stream. The possibility for unsupervised processing is now seen to be much closer. This is because the new set of assertions is easy to assess: we simply compare the predictions to the report stream and reward rules whose predictions are correct.**

4.2 Meta-level Rules for Implementing Environmental Payoff

Given the arrangement described above, construction of an unsupervised system can be elegantly achieved by the addition of a meta-level rule (a rule that mentions rules in its clauses) such as:

- ENV:** IF a rule predicts an occurrence and the occurrence is reported
 THEN
 increase the rule's wealth

The metarule implements environmental payoff of correct predictions by the explicit action of rewarding the rules responsible for the predictions. Notice that the rule is a second-order rule in two ways: (1) it mentions rules in its antecedants and consequents, and (2) the action it takes has a direct effect on a characteristic of the rule, its wealth, which is not a domain-level but system-level characteristic of the rule.

A useful elaboration of this rule computes a share of the system's overall payoff budget that the rule is to receive:

** A rule that predicts a report where there is no sensor coverage should not be viewed as having generated a failed prediction. To ensure stable operation, the system should pay such a rule the amount of its bid so that there is neither gain nor loss to the rule's wealth in such circumstances.

ENV2: IF a rule predicts an occurrence
and the occurrence is reported

THEN
increase the rule's wealth
by a specific proportion of the
total payoff budget

This rule has the virtue of implementing a closed economy (in terms of the overall wealth in the system). We are pursuing analysis of such closed systems to determine what convergence and stability results hold

Another way to achieve payoff strictly through the action of rules is to simplify the environmental rule as follows:

ENV3: IF a rule predicts an occurrence
and the occurrence is reported

THEN
report the occurrence with ENV as
the source

This rule demands that when a report is entered in the system, its source is given as ENV3. ENV3 is expected to be given a large share of wealth. When it is invoked it pays itself part of its bid (as the source of its second clause), and pays the rule that made the correct prediction the other part. This depletion of its wealth is made up for by the fact that when low-level sensor processing rules (e.g., Rules 1, 2, and 3, above) use reports, they pay their bids to ENV3. In this way, ENV3 implements the desired functionality strictly within the competitive bidding process described above.

Each of the metarules described implements unsupervised learning by driving the system towards sets of rules that make correct predictions. The result is exactly as in the supervised case described above, except that now a rule's wealth is proportional to its contribution to the system's overall predictive power. On the view implemented here, verifiable predictions are the only ultimate tests of correctness available to the system. In this system, utility is a completely internally assessed function of the system's ability to generate correct predictions about future enemy activities. It assesses its knowledge exclusively in those terms without a requirement for access or comparison to

5.0 SOURCES

Recall that the report stream is intended to reflect the situation where an artillery unit is stationed at some location, and a sensor is generating erroneous or bogus mover reports to the rule-based system. The system first believes it is in the presence of an armored unit, but soon begins to doubt its results since they consistently go unconfirmed. The system finally decides it is looking at an artillery unit. Thus the system achieves some level of adaptation in the face of an unexpected situation. That is not enough. A more complete system would not only assess the rules leading to the system's results, but also assess the sources involved in its processing, using

the source assessments, for instance, in re-evaluating its hypotheses or in collection management. We are exploring approaches to sensor evaluation by the removal of the all-encompassing "ENV"-type metarules and the introduction of an "environmental" rule for each external source of information available to the system. Such rules would look as follows:

Shooters: IF a rule predicts a shooting
activity
and the shooting activity is
reported

THEN
report the shooting activity
with "Shooters" as the source

Comms: IF a rule predicts a
communications activity
and the communications activity
is reported

THEN
report the shooting activity
with "Comms" as the source

Movers: IF a rule predicts movement-type
activity and the motion is reported

THEN
report the motion
with "Movers" as the source

Just as in the case of ENV3, the report sources, now defined as report-generating rules, accumulate wealth when sensor-processing rules use the source's reports and disburse payoffs when a rule correctly predicts a report emanating from the source.

Using the same sequence of reports as before, the effects of differentially accumulating wealth according to these three sources would be for each to accumulate wealth as their reports are used by Rules 1, 2, and 3. This would continue uninterrupted until the system decided it was looking at an artillery unit, at which time correct predictions would start being made. Once correct predictions begin, the sources, in this case Shooters and Comms, would dispense payoffs to Rules 7 and 8 analogous to how ENV3 would have dispensed payoff for correct predictions. Movers would keep its wealth intact. However, in this case Movers also would cease to play a part in the processing, since Rule1, which was the only rule using mover reports, starts losing the bid for the shooter reports to Rule2 (recalling the example at the beginning of Section 3.0). The source of mover reports has effectively been isolated from

5.1 Concluding Remarks

What are we to make of the wealths associated with sources? Movers' wealth was equal to that of Shooters and Comms and actually became greater than theirs once they began providing environmental payoff

to the predictive rules. Therefore we cannot interpret a sensor's wealth as a direct figure of merit. All the less so since we purposely set the stage with a false sensor stream from Movers! Is wealth a direct utility measure? This doesn't seem appropriate either, since the most useful thing about mover reports in this report stream is to find out that you shouldn't use them.

When a source fails, there seems to be an inverse relation between the wealth of source and its ultimate predictive power. Once predictions cease to be possible on the source, its wealth strictly accumulates until it is isolated. It appears that the wealth accumulated up to the isolation point of a source is simply a lost investment by the system. Possibly rules that make predictions about erroneous or misleading sources might lead to correct predictions, freeing this wealth for the system, but we have not yet experimented with this.

As we noted in the example above, inference paths that lead to failed predictions atrophy. In time other hypotheses are tried, and if some of these succeed they will isolate the erroneous source from further use by the system. If this does not happen, then either the inference paths effected continue to loose wealth or they form a self-supporting cycle of predictions in the system. The former is easily detectable, but it is computationally impractical to analyse the inference for such cycles. This suggests the importance of multi-a corrective to this type of problem.

We have presented an approach to the automated evaluation of hypotheses and information sources for rule-based systems. This work is being pursued at The MITRE Corporation in the context of a machine learning project (refs. 17-19). The dynamic rule evaluation is to guide the automated rule modification module of a system which will suggest new hypotheses about the domain to the user. We believe it will prove equally useful in interactive rule editing, where we expect the evaluation measures to guide the user to areas of the rule base needing modification, and in decision aids in the area of collection management, where a measure of efficacy for sources in the interpretation of a situation

REFERENCES:

1. R. P. Bonasso and H. J. Antonisse, "Expert Systems for Intelligence Fusion," Proceedings of the Army Conference of AI to Battlefield Information Management, Adelphi, MD, Battelle Columbus Laboratories, Washington Operations, pp. 101-116, April
2. M. A. Williams, "Distributed, Cooperating Expert Systems for Signal Understanding," Proceedings of the Seminar and Exhibition on AI Applications to the Battlefield, Ft. Monmouth, NJ, AFCEA, May, 1985, pp. 3.4-1 - 3.4-6
3. F. M. Atwater, "An Expert System Approach to Situation Assessment Using Sensor Fusion," Proceedings of the Seminar and Exhibition on AI Applications to the Battlefield, Ft. Monmouth, NJ, AFCEA, May, 1985, pp. 3.5-1 - 3.5-5.
4. The MITRE Corporation, "OB1KB: A Knowledge-Based System for Army Intelligence Analysis Using Order of Battle," K. B. Schwamb and E. Kasif, McLean, VA, MTR-86-W00133, Sept., 1986 (Not Available in the Public Domain)
5. F. D. Deffenbaugh, J. R. Miller, and L. E. Severin, "Knowledge-Based System for Tactical Data Reduction and Exploitation," Proceedings of the Seminar and Exhibition on AI Applications to the Battlefield, Ft. Monmouth, NJ, AFCEA, May, 1985, pp. 3.1-1 - 3.1-5.
6. T. D. Garvey and J. D. Lowrance, "An AI Approach to Information Fusion," Journal of Electronic Defense, 7:7 (July 1984).
7. D. B. Lenat, A. Clarkson, and G. Kiremidjian, "An Expert System for Indications & Warning Analysis," Proceedings of the Eighth International Joint Conference on AI, Karlsruhe, F.R.G., Aug. 1983, pp. 259-
8. S. J. Laskowski, H. J. Antonisse, and R. P. Bonasso, "ANALYST II: A Knowledge-Based Intelligence Support System," *Second Conference on AI Applications*, Miami, FL, AAAI, April, 1985.
9. S. J. Laskowski and E. J. Hoffmann, "Script-based Reasoning for Situation Reasoning," Proceedings of AAAI-87, (Seattle, WA), AAAI, July 1987.
10. J. W. Benoit, J. R. Davidson, E. J. Hofmann, S. J. Laskowski, and R. R. Leighton, "Integrating Plans and Scripts: An Expert System for Plan Recognition," Proceedings of the 3rd Annual Expert Systems in Government Conference, Washington, D.C., IEEE, October, 1987.
11. J. McDermott and C. Forgy, Production System Conflict Resolution Strategies, in: "Pattern-Directed Inference Systems", D.A. Waterman, F. Hayes-Roth (eds.), Academic Press Inc., New York, (1978).

12. M. Minsky, "Steps Toward Artificial Intelligence", in: "Computers and Thought", A.E. Feigenbaum and J. Feldman (eds.), McGraw Hill, New York (1961).
13. A. L. Samuel, "Some Studies in Machine Learning using the Game of Checkers", IBM J. of R&D.
14. J. H. Holland, "Properties of the Bucket Brigade Algorithm", Proceedings of the First International Genetic Algorithms Conference, Lawrence Erlbaum, Englewood Cliffs, NJ (1985).
15. J. H. Holland, "Adaptation in Natural and Artificial Systems", University of Michigan Press, Ann Arbor (1975).
16. R. Riolo, "Bucket Brigade Performance", in: "Genetic Algorithms and their Applications", J. Grefenstette, (ed.), Lawrence Erlbaum, Englewood Cliffs, NJ (1988).
17. H. J. Antonisse, K. S. Keller, and H. Tallis, "Adaptive Techniques for Rule-based Systems: Rule Evaluation", MITRE Technical Report MTR-86W00161, October, 1986 (Not Available in the Public Domain)
18. H. J. Antonisse and K. S. Keller, "Dynamic Evaluation of Imprecisely Specified Knowledge", Proceedings of the 7th Annual Digital Avionics Systems Conference, Ft. Worth, TX, May, 1986.
19. K. S. Keller and H. J. Antonisse, "Prediction-based Competitive Learning in the M2 System", Proceedings of the 3rd Expert Systems in Government Conference, Washington, D.C., October, 1987.

INTEGRATION AND EXPERIMENTATION WITH COOPERATING EXPERT SYSTEMS, NATURAL
LANGUAGE PROCESSING, AND ALGORITHMIC TECHNIQUES TO PERFORM CORRELATION
AND DATA FUSION IN A TACTICAL ENVIRONMENT

James B. Nelson

Lockheed Austin Division
O/T0-10, B/30F
6800 Burleson Road
Austin, Texas, 78744
(512) 448-9488

Existing Tactical C3I Centers are limited in their ability to process (accept, analyze, correlate, fuse, display, and manage) the voluminous multiple-format, multisource message data that are input to the center. This limitation will become worse as new sensors and communications improvements become operational. C3I Centers of the future will be severely impacted by these changes, and, as a result, will require more automation and processing sophistication.

This paper presents the results of research and development work that has focused on rapid-prototyping and experimenting with advanced processing methodologies to support C3I Centers of the future, particularly those that require intelligence data correlation/data fusion and operational use of those data in a timely manner. Four technologies applicable to C3I Centers are addressed in this paper.

1. Automatic Message Processing and Correlation: Two algorithmic correlators, one each for communications and electronics data, were implemented to create and update an intelligence Order-of-Battle Data Base. These correlators are based on statistical parameter and attribute matching techniques. They function on parsed and filtered JINTACCS messages and are under analyst control, yet perform automatically.
2. Cooperating Expert Systems: Three cooperating expert systems are used to examine the Order-of-Battle Data Base created by the previously described algorithmic correlators. These are a) Correlation and Association Enhancement, b) Force Structure Recognition and Tracking, and c) Activity Monitoring. These expert systems use knowledge bases, stored in the form of semantic network of frames, in rules and in computational functions. The knowledge base contains doctrinal data, emitter characteristics, and analytical heuristics. An endorsement method is used to reason with uncertainty.
3. Natural Language Text Line Processor: The natural language technique uses a case frame grammar to parse the JINTACCS messages syntactically and interpret them semantically. This processor provides recognition and expansion of acronyms and abbreviations, interpretation of free form text and multiple text lines in the same message, prompts the analyst, and formats the results into frames for use by the expert systems.
4. Weaponing/Targeting Functions: Algorithms and analyst interfaces were developed and implemented to provide automated target-weapon pairing, target-weapon scheduling, and use of the Joint Munitions Effectiveness Manual methodology. These algorithms operate automatically against the intelligence Order-of-Battle Data Base and perform the following functions: calculates optimum allocation of weapons to an array of targets, determines optimum allocation of loaded weapons to aircraft, calculates single-sortie probability of damage.

These technologies are integrated into a C3I Testbed System that is used to demonstrate and evaluate the application of advanced processing techniques to the domain of tactical intelligence processing, analysis, and to timely use of those data by a tactical battle manager.

INTEGRATION AND EXPERIMENTATION WITH COOPERATING EXPERT SYSTEMS, NATURAL
LANGUAGE PROCESSING, AND ALGORITHMIC TECHNIQUES TO PERFORM CORRELATION
AND DATA FUSION IN A TACTICAL ENVIRONMENT

James B. Nelson

Lockheed Austin Division
O/T0-10, B/30F
6800 Burleson Road
Austin, Texas, 78744
(512) 448-9488

INTRODUCTION

Many existing tactical command, control, communications, and intelligence centers are severely limited in their ability to process automatically the large volume of multiple-format, multisource message data they receive. Currently fielded systems lack the processing sophistication that is needed to reduce this voluminous amount of data into useful, manageable information. This shortfall in processing capability of current systems has stimulated the development of new, improved, state-of-the-art processing techniques that are more accurate, save time, and provide the battle manager with useful, reliable, automated tools.

Purpose

The purpose of this project was to investigate, develop, demonstrate, and evaluate the application of advanced processing techniques (automatic message processing, algorithmic correlation/data fusion, natural language processing, expert systems, and relational data base interfaces) to the domain of tactical intelligence processing and analysis and to examine the timely use of those data by a tactical battle manager.

Experimentation

Five specialized vignettes were created (by domain experts) and added to a 1990s Fulda Gap war scenario. These were input to the Testbed system (1) to determine the effectiveness of the algorithmic and heuristic approaches to correlation/data fusion, (2) to observe how well the Natural Language Text Line Processor (NLTLTP) and Expert Systems (ES) performed, and (3) to ascertain if a dynamically created and updated intelligence Order-of-Battle (OB) could be used automatically by a Weaponizing (Targeting) function of a tactical C3I center.

Vignette I. Delineation of Unit Boundaries. This vignette is structured to capture the knowledge an analyst applies when trying to determine the area of interest of a threat division, the forward line of troops, the boundaries between forces adjacent to the division, and the boundaries of subordinate regiments. Recognition precepts include inferences on air defense assets, critical node location, and artillery command and control.

Vignette II. Recognition of Air Defense Artillery Assets. This vignette focuses on tactical deployment of division air defense assets: their electronic signatures and communications links. It deals with the thought processes that an analyst uses to identify those assets, locate them, and track them. Recognition precepts include target acquisition battery inferences, echelon and organizational inferences, relative location inferences for emitters, and overall deployment inferences.

Vignette III. Identification and Location of Command Posts. This vignette is designed to activate the thought process of the analyst when he/she identifies, locates, relates, and interprets the significance of Command Posts. Recognition precepts include signatures, geolocations, organizational data, interactions, and tactical inferences.

Vignette IV. Recognition of Attack Activities. This vignette provides the stimulus necessary to activate those ES rules that deal with recognition of the type of combat activity about to unfold. Recognition precepts include communications activity, special equipment, frontage size, and asset deployment inferences.

Vignette V. Enhancement of Algorithmic Correlation. This vignette is designed to stimulate the refinement, assessment, and accuracy of OB Database files of emitters and units. It permits the ES, which contains multiple hypotheses about emitters or units, to recorelate questionable decisions made by the algorithmic correlator.

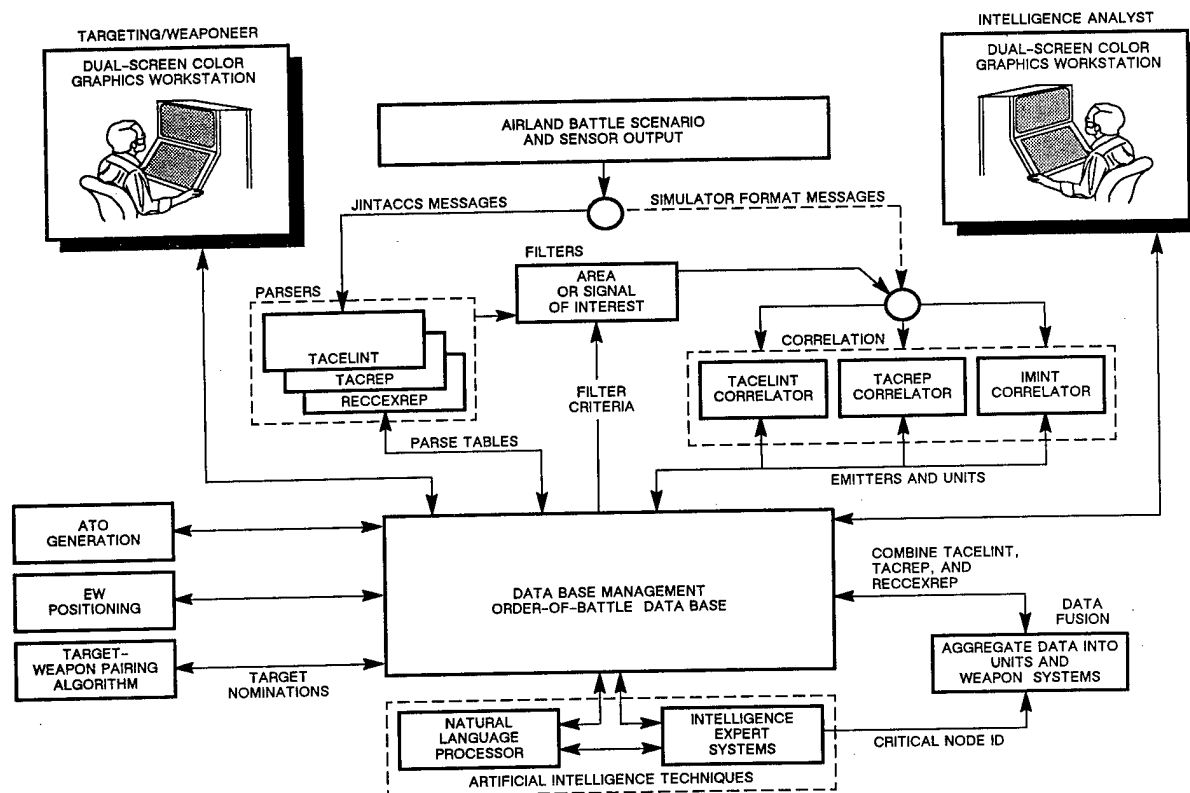


Figure 1. Integrated Testbed Architecture

SYSTEM ARCHITECTURE

The overall system (C3I Testbed) is a fully concurrent, multiuser information processing system where processing is divided into functional blocks with well-defined interfaces between modules. The Testbed functional architecture (Figure 1) permits experimentation and demonstration of integrated technologies as applied to the intelligence and weaponeering functions of a tactical C3I center.

Fully formatted JINTACCS messages are input to the Testbed by an AirLand Battle simulator, which uses a 1990s Fulda Gap war scenario (enemy events and OB) and a variety of airborne sensor models. The Testbed automatically parses (decodes) and filters the JINTACCS messages according to tables that are set and easily changed by the Intelligence Analyst. Two algorithmic correlators (statistical and figure-of-merit based) perform automatically to read the message data and to create and update emitter files within an OB Database.

A Data Fusion function examines the emitter files created by the correlators, adds the information produced by the artificial intelligence (AI) techniques, then combines the data to form units and weapons systems within the OB that represents the most complete information about entities on the battlefield.

The AI techniques include the NLTLTP and distributed cooperating ES. The NLTLTP reads and interprets the English language text lines of the JINTACCS messages, reformats the data, and passes the results to the ES. The ES examines the OB Database and the NLTLTP output to determine the presence of certain types of units on the battlefield, their geographic operating boundaries, and activity state. Additionally, the ES recognizes and corrects certain types of errors not handled by the algorithmic correlators.

The Weaponeering/Targeting function automatically accesses the intelligence OB Database, compiles target lists, and calculates target-weapon pairing solutions based on a global optimization algorithm.

An imagery processing capability is in the planning stages at this time.

An Air Task Order (ATO) generation capability is in the development stage.

The system architecture permits a wide range of experimentation possibilities for the testing of specific modules, algorithms, or fully integrated components.

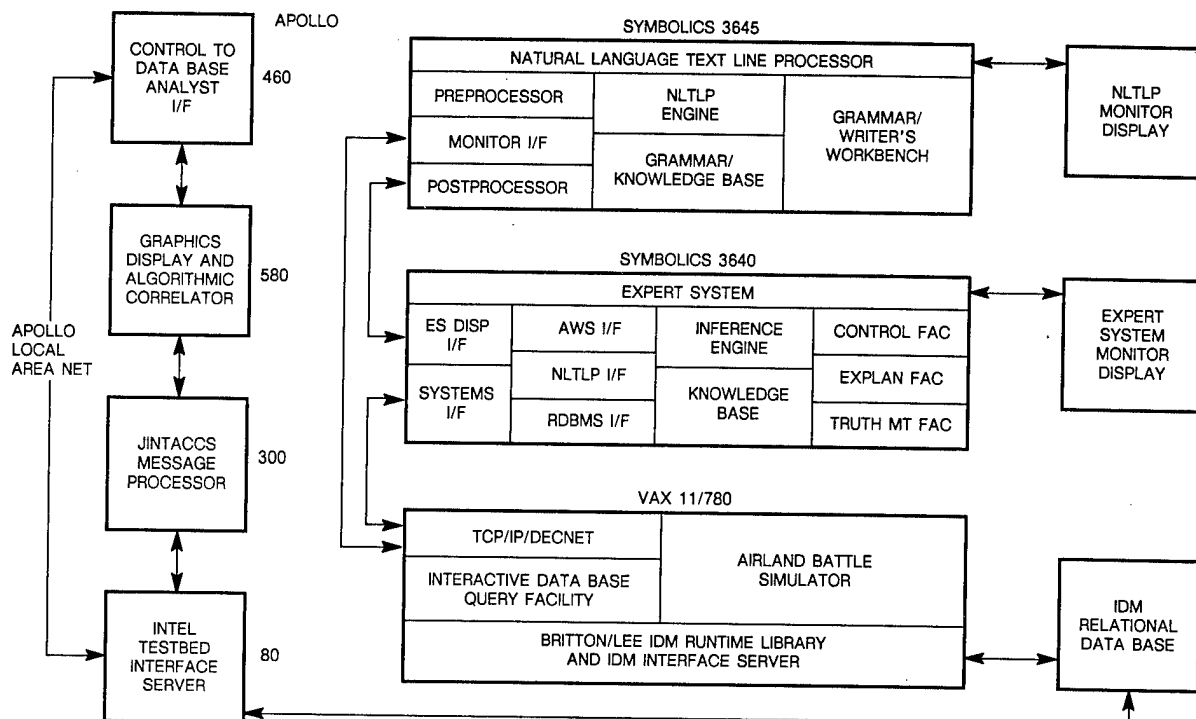


Figure 2. Hardware/Software Modules and Interfaces

Hardware/Software Implementation

The approach used to implement the C3I Testbed System is characterized as a balanced integration of conventional processing technology with AI processing techniques in a distributed environment. Each technology is applied to the intelligence analysis and targeting problems in the manner deemed most appropriate. This approach does not attempt a total AI solution nor does it attempt to force AI techniques on aspects of the problem that have already been solved by algorithmic methods. Figure 2 shows the hardware connectivity and the software major module integration of conventional algorithmic processes with AI heuristic processes.

Distributed Processing Interfaces

This project emphasized the use of distributed processing techniques as a way to achieve fast and useful interfaces between the various workstations and processes. These interfaces permit the ES, NLTP, algorithmic correlator analyst displays, Weaponeer, and data base management functions all to reside on separate processors, yet be accessible by any. Two primary classes of interfaces are used: RS232 software interface and TCP/IP software. In addition to vendor-developed TCP/IP system software for specific machines, we developed a remote procedure-call applications layer for interface between the Apollo and Symbolics machines.

Software Tools and Programming Languages

The Testbed System uses a variety of software techniques to make the system work. These are summarized as follows:

Simulator: Ada, Vax 11/780

Expert Systems: Common Lisp, Automatic Reasoning Tool (ART) ES Shell, Symbolics. ART was chosen for its graphics, rule definition and compilation, confidence levels, and viewpoints (blackboard) capabilities.

Natural Language: Common Lisp, Language Craft NLP Shell, VAX, and Symbolics. Language Craft was chosen because it is a general-purpose, natural-language processing tool, it permits the developer to concentrate on vocabulary and semantics rather than mechanics of parsing and semantic interpretation, it has a well-defined interface to Common Lisp, and runs on both VAX and Symbolics.

Map Graphics: C, Apollo (also runs on Silicon Graphics, Sun, MicroVAX). Digital map graphics display technology (SoftCopy Map Display System) contains DMA data (DTED, DFAD) and World Data Base II and has user friendly software that permits extensive map manipulation and control.

User Interface Manager: C, Apollo (also runs on Silicon Graphics, Sun, MicroVAX). System permits creation/change of displays in minutes by a nonprogrammer.

Correlators: FORTRAN, Apollo.

Data Base: IDL and Progress Relational DB Access, Britton-Lee.

Message Processing: FORTRAN, Apollo.

ALGORITHMIC CORRELATORS

Both the Tactical Electronics Intelligence (TACELINT) and the Tactical Report (TACREP) correlation algorithms function automatically within the Testbed message-handling architecture. Each has a set of specialized interactive tools that permit an analyst to control automatic correlation variables: review, edit, or change the results of automatic correlation and make manual correlations.

TACELINT CORRELATOR

The TACELINT Correlator performs the following primary functions:

- Attempts to correlate signal-of-interest messages to existing emitter OB Data Base files automatically
- Updates emitter OB Data Base files automatically
- Sends graphic updates to the map graphics display (analyst workstation).

The TACELINT algorithm is based on a scoring and weighting system that calculates the statistical differences between the emitter parameters contained in a message and the parameters of emitters held in the data base that fall within a tolerance gate set by the intelligence analyst. Those OB Data Base emitters that fall within the tolerance gate are considered to be reasonable candidates for correlation. Each parameter of each emitter is scored and weighted (by any criteria as determined by the analyst, e.g., sensor measurement, time delay, etc.). Next, each candidate is tested for geospatial feasibility and scored. Total scores are then calculated for each candidate versus the message emitter. Finally, the algorithm compares each candidate. If a single best candidate (heuristically determined by the analyst and input to the algorithm) is identified, a correlation is made. Otherwise, candidates are ranked and filed in the OB Data Base as potential (ambiguous) correlations. Each ambiguous emitter is reconsidered as new messages are input to the TACELINT correlator.

TACREP CORRELATOR

The TACREP correlator uses parametric association and attribute-matching techniques to correlate message data with emitters held in the OB Data Base. Automatic correlation occurs when certain combinations of data elements in the message match exactly or within tolerances with the data elements contained in the data base. If a unique identifier is present in both the message and the OB Data Base, if the candidate meets a geospatial feasibility test, and if the candidate meets a scoring threshold (set by the analyst), a correlation is made. Those attempts at correlation that do not meet the minimum score result in the message data being returned to a message file for future attempts at correlation. The TACREP correlator does not create ambiguities.

DATA FUSION

The Data Fusion function uses the emitter OB Data Base files, previously created by the TACELINT and TACREP correlators, to create and maintain weapons systems and units in the OB. It also uses the data provided by the ES and the analyst workstation.

Data Fusion performs automatically, making inferences about the battlefield. At the first level, inferences proceed directly from a single fact, e.g., the existence of a weapons system can be inferred directly from the existence of certain emitters. Some unit types can also be inferred in this manner. A second level of inferences occur when inferences are made based on several facts that do not imply anything separately, but when considered together imply something about the OB. For example, the existence of several different entities, located within a certain distance of each other, or the determination of a parent-child relationship between two or more entities, can imply much more than the existence of a single entity.

The Data Fusion function performs the following functions:

- Groups entities into systems
- Determines battlefield positions for systems based on locations of component parts
- Deduces the presence of units from the deployment of specialized equipment
- Assigns systems and equipment to the controlling unit
- Determines battlefield unit positions
- Determines unit organizational hierarchy.

The Testbed System uses tightly coupled OB Data Base, Correlation, and Data Fusion functional modules as a complete OB management system. Each of the automatic processes is responsible for maintaining its specialty data in the central data base and on the map display. All changes to the OB, whether initiated by the analyst, ES, Data Fusion, or Correlator, maintain the integrity of the OB Data Base. For example, the parametric details of TACELINT correlation may be unknown by Data Fusion, but if the emitter had moved, the communication of that fact to Data Fusion would cause the Data Fusion algorithm to recalculate the position for the unit containing that emitter.

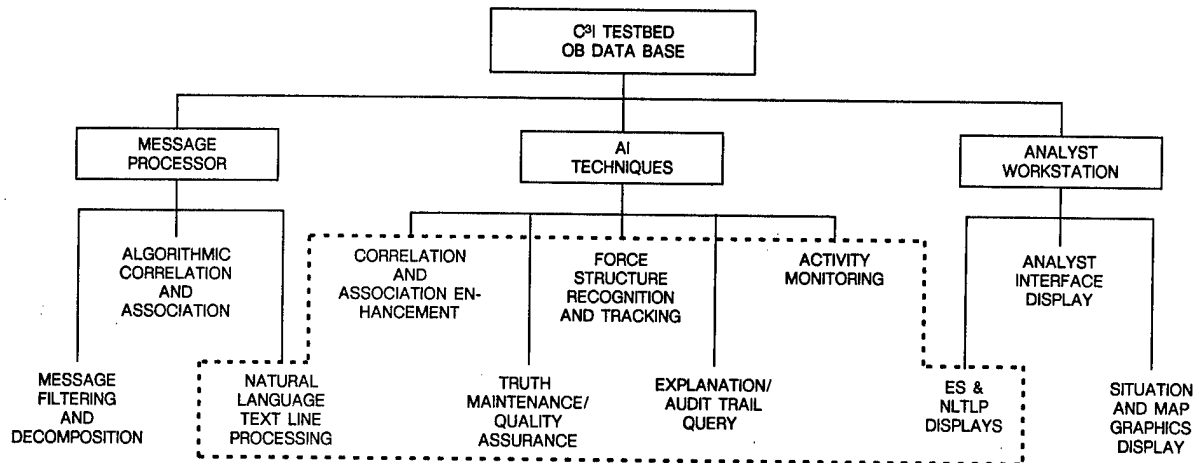


Figure 3. AI Techniques Integrated with Testbed System

ARTIFICIAL INTELLIGENCE TECHNIQUES

Seven component subsystems using AI techniques were integrated into the Testbed System. These components are organized as illustrated in Figure 3. The ES includes three cooperating subexperts and two global support functions.

Correlation and Association Enhancement

The Correlation and Association Enhancement (CAE) subexpert looks for constraint violations in the output of the algorithmic correlators, directs the correlators to reassociate the associated messages, and communicates with the correlator in a manner functionally similar to that of the human analyst. The CAE works in conjunction with the FRT to perform constraint-based correlation and data fusion. Constraint-based correlation is performed using constraints along one or more of several different problem space dimensions (or classes): Parametric, Geopositional, Doctrinal, Terrain, Activity, and Multisource. CAE is closely coupled with the algorithmic correlator and has a feedback loop with it to pass information on reassociation and change of OB Data Base.

Force Structure Recognition and Tracking

The Force Structure Recognition and Tracking (FRT) subexpert accomplishes the following: hypothesizes force structures from the association of emitter reports; determines the types of units represented using doctrinal, table of organization and equipment (TO&E), and analytical knowledge; tracks the movement and confidence associated with the force structures and units. The FRT formulates and compares the positional clusters formed by individual emitters, determines subclusters, and examines the relationships between clusters to form an opinion of the existence of certain types of units. This subexpert also compares the total OB known to exist within an area of interest to doctrinal data to determine if there is agreement between the perceived and doctrinal OB. If discrepancies are found, they are reported to the analyst. In addition, the FRT performs specific unit identification, including echelon and subordination.

Truth Maintenance

The Truth Maintenance Facility maintains the interrelationships between objects and permits the system to backtrack whenever a given object characteristic or occurrence is found to be in error or is contradicted. These interrelationships are derived from the endorsements of each object maintained for reasoning under uncertainty and allow for chaining from one object and fact dependency to another.

Explanation Facility

The Explanation Facility provides an English-like explanation of the reasoning process behind each decision made by the three subexperts. This facility responds to analyst requests for explanation regarding any emitter, unit, or activity indicator. The explanations are derived from the endorsements of each object maintained for reasoning under uncertainty and include an audit trail of the inferences performed rather than just a list of rule names. Each rule included within any of the ES subexperts is coded with a brief English-language explanation of the rule function, and when a rule fires, an endorsement is created for the object of interest and is stored with the object. By using the accumulated endorsements as the basis for explanations, a separate audit trail does not have to be maintained and overall system consistency is ensured.

Activity Monitoring

The Activity Monitoring (AM) subexpert recognizes activity indicators associated with the units in the OB, accumulates warning signs, and posts alerts to the analyst for indicators of attack. It uses the output of the Natural Language Processor and the output of the algorithmic correlators. Activity indicators are accumulated by activity category and by unit type involved. The AM uses a hierarchical structure of indicators, warnings, and alerts so that any type of activity can be included. If multiple indicators occur for a given activity or unit type, a warning is provided to the analyst. If multiple warning are created regarding a particular activity type, an alert is created and issued to the analyst.

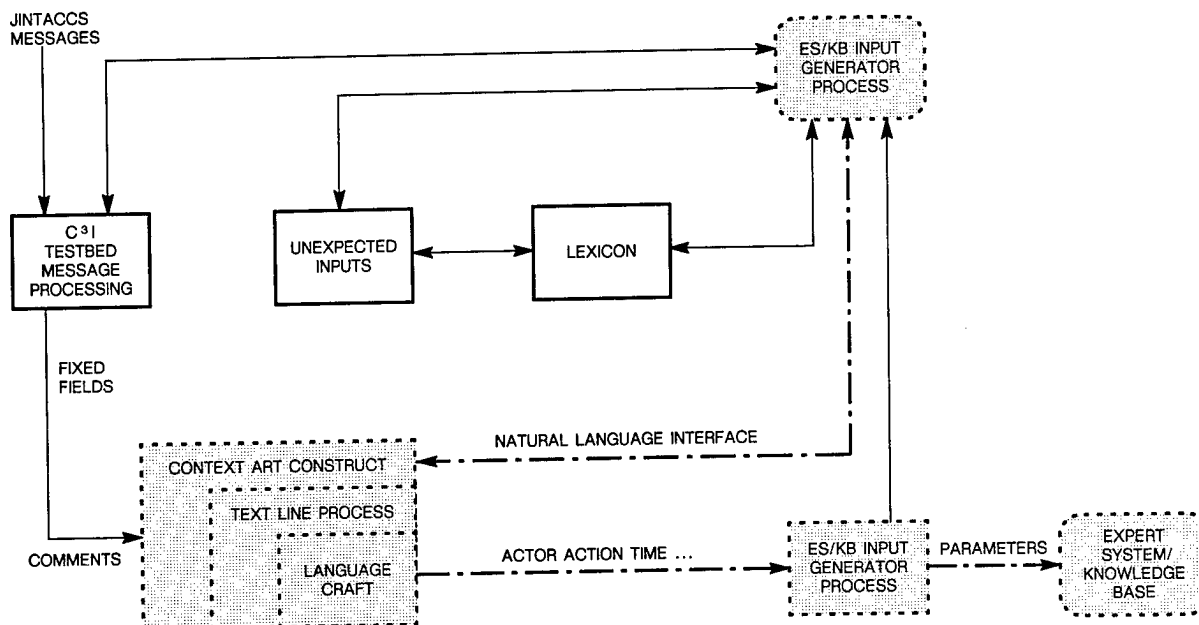


Figure 4. Natural Language Functions Integrated with Testbed System

Natural Language Text Line Processor

The NLTLP was designed to combat the difficulties of interpreting free text and creating a usable form of data for the ES. The most important goals were to examine the techniques needed for processing free text and to create an automatic process to accomplish this. The techniques explored were the caseframe grammar used, along with discourse analysis techniques such as using a context mechanism, to complete missing values. These techniques were written in Common Lisp, along with the grammar written for use with the Language Craft parser. The techniques were initially implemented as an automatic process; however, an entirely automatic process does not appear to be the best solution because ambiguous (multiple) parses can occur unless the grammar is constructed very carefully. Ambiguous parses can be presented to the analyst to resolve, saving computer time. The integration of the NLTLP within the Testbed System is shown in Figure 4.

NLTLP Functions and Design

The Natural Language Text Line Processor accepts the fixed fields and free-form text lines of both TACELINT and TACREP messages, parses the free-form text, and interprets the meaning of this text in the context of the fixed fields and other text lines. The results of this processing are records that can be processed and interpreted by the ES. The NLTLP includes preprocessing functions to handle abbreviations, acronyms, and simple misspellings. It handles a variety of activity types, verbs, nouns, and modifiers. It can distinguish tenses and can correctly interpret time and place information. The NLTLP uses a caseframe grammar approach to parsing, and semantic interpretation makes the NLTLP easy to extend to new reports, activities, objects, and modifiers.

Preprocessing of Free Text

Preprocessing is done on text lines before they are parsed by grammar. Several preprocessing steps are used:

- Message handling with and without text
- Breakdown of multiple text lines into individual text lines
- Preparation of each text line (acronym expansion, separation of numbers and characters in time and location fields, addition of case markers to enhance parsing, and handling of abbreviations not allowed by Language Craft).

Components of Grammar

The grammar used for the NLTLP processing is the caseframe grammar that is used with the Language Craft parser (Plume), consisting of the four main components: caseframes, rewrite rules, irregular morphology rules, and abbreviation rules.

Postprocessing Rule

Postprocessing rules operate on the instantiated caseframe generated by the parsing of a text line. Three forms of postprocessing rules occur:

- Mapper Rules. These are used to transform the caseframe instances output by the parser into caseframes for input to the LISPIfier rules.
- LISPIfier Rules. These are used to convert mapper output into LISP forms suitable for a particular application.
- Context Resolution Rules. A context schema is created from analyzing each text line, and if a resultant schema is missing information, previous schemas are checked to fill the missing values.

Figure 5 provides an overview of the NLTLP functional flow.

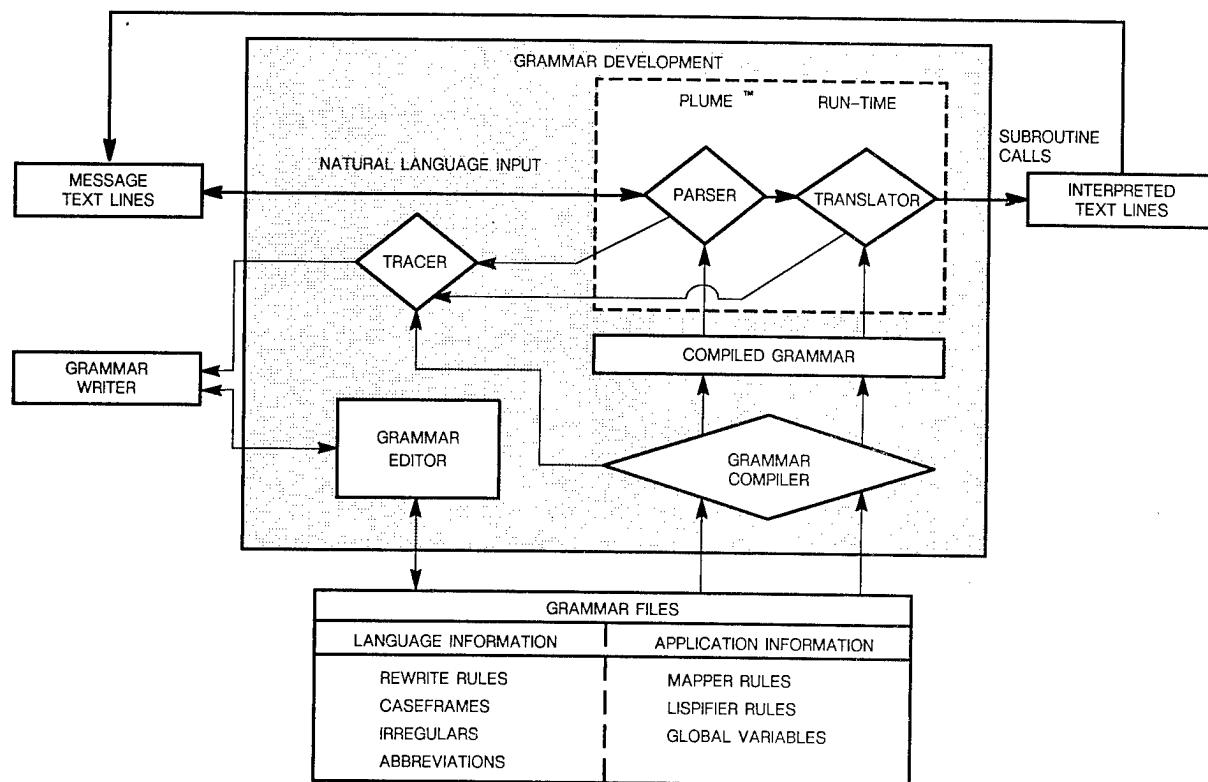


Figure 5. Natural Language Text Line Processor Functional Modules

AI Techniques Control Mechanisms

To achieve a balance between control (reduce the time spent responding to a particular input) and flexibility (spontaneous handling of expected and unexpected inputs), the ES design consists of the following: cooperating subexperts, use of an endorsement method for reasoning under uncertainty, and a mixed-initiative user interaction.

- **Cooperating Subexperts:** Primary control was established in the FRT subexpert because algorithmic correlators have already processed much of the data. However, the system uses trigger events to assist in transitions between subexperts so that certain events cause primary control to switch to the appropriate subexpert.
- **Endorsement Method:** This method captures endorsements or votes for or against a particular conclusion, which is then reviewed, summarized, and assigned a confidence level. As votes are collected, a record of the support for the typing of each object is kept, and the evidence is increased or decreased as appropriate. Endorsements also serve as a basis for constructing explanations of system decisions and for truth maintenance.
- **Mixed-Initiative User I/A:** This approach to ES control allows both the analyst and the ES to direct the flow of processing.

Relational Data Base Interface

Relational data bases provide an especially flexible organization system for storing large volumes of data and data relationships in tabular form. Interfaces between relational data base management systems and AI systems, like the NLTLP and ES, permit the latter to retrieve data and information automatically and selectively and store the results of the processing for later use by both systems and operators.

The relational data base interface allowed the NLTLP and ES to share a common relational data base for the OB with the algorithmic correlators and the analyst interface. It allows the ES to submit data base queries autonomously and update requests to the management system.

Analyst-Machine Interface

The project demonstrated the use and integration of both AI and conventional technologies for the analyst-machine interface. The interface includes a menu-based query-and-command, interactive map graphics, and an ES monitoring-and-control function. For the ES, the analyst can ask for descriptions of objects in the OB and explanations for their existence. The correlation analyst can query, move, change, and ask for unit hierarchy or parametric data, uncertainty ellipses, unit or emitter histories as well as perform other analytic functions.

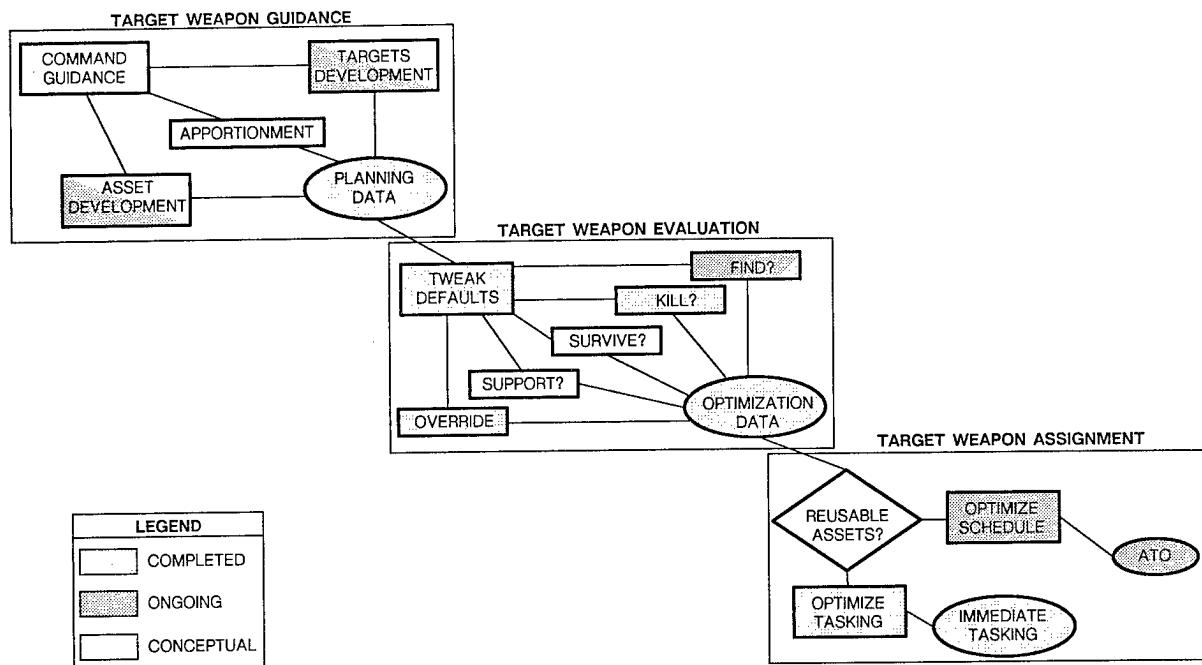


Figure 6. Target-Weapon Decision-Aiding

WEAPONNEERING/TARGETING FUNCTIONS

The Testbed System intelligence analysis functions were integrated with a battle management function: targeting and weaponneering. This interface permits the battle management functions to access a dynamically created and updated intelligence OB, as it is happening. During system tests, we were able to observe the input of message-containing emitter data from the simulator into the Testbed Parser and Filter, Correlator, Data Fusion, through the NLTLP and ES, into the OB Data Base, and finally, watch the data appear as a target at the weaponneering/targeting analyst display.

Target-Weapon Pairing

The Target-Weapon Pairing (T-WP) algorithms determine the optimum allocation of weapons systems on alert (or with short response times) to an array of targets that have been selected. The allocation is based on the OB, weapons information from an Assets Data Base, the Single-Sortie Probability of Damage from the Joint Munitions Effectiveness Manual (JMEM), and the analyst's input of weights to be given for such factors as attrition, weapon system cost, and scarcity. The T-WP algorithms pair available weapon resources to a list of prioritized targets based on a criteria function, the number of weapons available, and the number of weapons needed to attack at the desired level of effectiveness. The weapons/targeter analyst can chose targets on the OB map display using the cursor or by inputting target identifiers. Solutions are automatically generated and displayed. They may be saved while new criteria are entered for a different solution, compared, and either rejected or accepted. The algorithm handles up to 25 targets simultaneously to pair up with 12 weapons systems standard conventional loads.

Target-Weapon Scheduling

The scheduling algorithm determines the best way to load and assign weapon systems to all targets in a given Air Task Order (ATO) generation cycle. It compares available weapon assets with a target list based on multiple sorties per weapon and the cost effectiveness of all target-weapon pairs.

The algorithm maximizes the sum of the cost-effectiveness values for the target-weapon pairs subject to the following constraints: number of available weapons, the minimum time from take-off for an aircraft from a given weapon set, number of sorties an aircraft can accomplish in one day and its cycle time, windows for each target, and weather for both the aircraft home base and the target. The approach avoids the trap of assigning the best weapons to the highest priority targets by detecting anomalies or special cases in the total target-weapon-time matrix for the scheduling period.

Joint Munitions Effectiveness Manual

JMEM methodology, along with user functions that permit methodical and standalone execution of the algorithm, were implemented in the system. The JMEM algorithm calculates the weapon-effectiveness estimates of the single-sortie probability of damage and number of weapons needed to achieve the probability of damage for a given target type and weapon combination.

Target-Weapon Pairing System: Aided Decisions

Under development within the Battle Management laboratory is a rapid prototyping and evaluation of decision-aids that would further automate the battle management functions of tactical C3I centers of the future. Figure 6 illustrates a part of the decision-aiding necessary to advance the automation of this cycle.

Many existing tactical command, control, communications, and intelligence centers are severely limited in their ability to process the large volume of multiple-format, multi-source message data automatically. Currently fielded systems lack the processing sophistication that is needed to reduce this great quantity of data into useful, manageable information. This shortfall in processing capability of current systems has stimulated the development of new, improved, state-of-the-art processing techniques that are more accurate, save time, and provide the battle manager with useful, reliable, automated tools.

RESULTS AND CONCLUSIONS

The most significant operational conclusion regarding the application of artificial intelligence technologies to tactical intelligence processing and analysis is that these technologies, including the ES and NLTL, can significantly enhance the support provided to the analyst over conventional methods if they are applied to the more abstract and less quantitative portions of the problem domain. These capabilities complement proven conventional and algorithmic approaches and provide an intelligent assistant for the analyst in the areas of correlation enhancement, text processing, and intelligent data base interface. The NLTL and ES provide a faster method of handling high volumes of messages because they can reason about incomplete and uncertain data, provide inferences at multiple levels, and maintain multiple contexts.

The Testbed System demonstrated that analyst expertise can be embedded in a system. The ES knowledge base includes the doctrinal, TO&E, parameter data, and analytical expertise used by the analyst to process tactical intelligence reports. The ES inference engine uses this knowledge to perform analyses, reason about uncertain data, and present (display) the current tactical situation. The NLTL knowledge base includes the grammar, vocabulary, and contextual associations that must be understood by an analyst to interpret these reports.

The use of a relational data base interface significantly reduced the analyst workload by automatically retrieving and updating data and information from data bases for use by the ES and NLTL.

Natural language processing can extract useful information from free-form text, associate this information with the formatted data (report context), and format it for use by the ES. The NLTL processes messages much more rapidly and accurately than human analysts. The use of caseframes to support natural language processing permits smooth integration of an NLTL and ES.

Advanced and conventional techniques can be interfaced, can use a common analyst-machine interface, and can be implemented in a distributed environment that shares a common data base.

The development of customized knowledge bases, expert systems, natural language processors, and interfaces for

each user or for each tactical unit is not necessary. A generic model of knowledge and metaknowledge can be developed that will serve as a basis for supporting the majority of tactical analysis needs.

Most specific knowledge in a knowledge base can be stored in frame structures and general rules rather than using specific rules. The frame, a flexible record-like structure with slots (fields), relationships, and attached procedures, can be used to represent most of the specific knowledge in a knowledge base, and general rules can be coded that reference entire classes of patterns rather than very specific patterns and data values.

A single deterministic method (endorsement) can be used as the basis for reasoning under uncertainty, truth maintenance, and explanation.

The Testbed System demonstrates a prototype of interoperability and integration of intelligence analysis (preparation of the OB) and weaponizing (targeting) functions of a tactical C3I center. This prototype indicates a significant time savings in generating targeting solutions through the use of an automatic optimization scheme and the use of direct interface with the intelligence OB Database.

FUTURE DIRECTIONS

Future research issues concerning the use of advanced processing techniques for tactical C3I centers can be outlined as follows:

- Automated knowledge acquisition and knowledge base extensibility, particularly for field use
- Cooperating distributed ES
- Extended natural language processing with human interfaces and contextual explanation generation
- Sensor-cueing based on situation assessment stemming from ES such as the Activity Monitoring and Force Recognition and Tracking
- Terrain-based geofeasibility extension to enhance correlation accuracy
- Extended automation and netting of Battle Management functions with intelligence functions.

DATA FUSION AS THE GUIDING PRINCIPLE IN THE BROAD AREA SEARCH PROBLEM(U)

Dr. R.H. Cofer

Government Information Systems Division
Harris Corporation
P.O. Box 98000
Melbourne, FL 32902

(U) The problem of keeping track of movable targets operating over broad areas of terrain poses a significant challenge to the intelligence community. Intelligence data in a variety of forms from a variety of sensor platforms is available, but the time interval between observations of targets is likely to be such that the target positions will change; perhaps significantly, during the intervals. Sensor observations of the target area can provide at least partial information about the operational status of the target. This information should be fused with ancillary information about road networks, terrain features, weather factors, deployment tactics, and constraints on physical movement. In addition there is a need for integration of the sensor tasking processes to insure the best possible surveillance of the broad areas.

(U) This paper describes an expandable framework whereby an automated fusion tool can be constructed to assist in the solution of the broad area search problem. The framework is analogous in some ways to the Kalman-filter approach to state estimation problems: There is a predictive component that uses all available information to produce continuous-time estimates of the locations of targets, and there is an observation component that provides new information at non-uniform discrete intervals of time. These two kinds of information are fused to produce an improved estimate of the locations of targets, and to guide future observation times and places. This is done by an algorithmic inference engine which propagates the estimate of the state of the system (the target location) in a way that keeps track of the multiple possibilities of the new unobserved location of targets that moved. Probabilistic confidence levels are associated with each possible predicted location, and sensor tasking management is then based on maximizing the chance of finding a moved target. The propagation process is computationally intensive but well within the state of the art. Simulation photographs of the process taken during a scenario are provided along with a description of the underlying framework.

DATA FUSION
AS THE GUIDING PRINCIPLE IN THE BROAD AREA SEARCH PROBLEM (U)

Dr. R. H. Cofer

Government Information Systems Division
Harris Corporation
P. O. Box 98000
Melbourne, FL 32902

(U) INTRODUCTION

(U) Figure 1 shows a typical intelligence process reduced to fundamentals. A movable target is acting out its own scenario oblivious of any specific sensor tasking or observation. It however generally realizes the potential for observation and divides its time between deployment and broad area relocation to avoid detection.

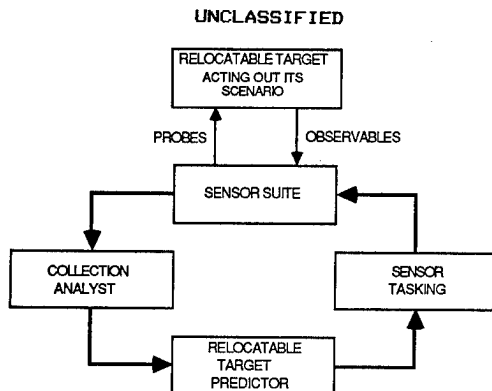


Figure 1. (U) The Intelligence Processing Cycle

(U) The intelligence process would like to maintain contact of the target but is hampered by the lack of continuous observation. The obvious question is where to place the next sensor collection to maximize the chance of picking up the evading target again. This leads naturally to the problem of fusion of observation and target prediction functions in order to properly task available sensor resources.

(U) Solution requires due attention to proper use of disparate informations: terrain mobility, target maneuverability and

doctrine, opportunistic intelligence reports, and sensor availability: all as discontinuous functions of time and space. The key to the solution is maintenance of a Kalman-filter analogous set of target state probabilities.

(U) The following three sections discuss the algorithmic inference engine concepts used, Figure 2. Then two overview examples are provided to point out features of its use. The principal conclusion to be drawn is that the techniques proposed can provide a significant aid to the overall intelligence process.

(U) TERRAIN INTERACTIONS

(U) During lack of incoming intelligence, the only information available is that of terrain influences on target maneuverability and velocity. To simplify the computer solution, terrain locations are expressed as points, (i,j) , on a square digitization grid.

(U) Maneuverability

(U) The terrain holds great influence over the target's maneuverability and deployment patterns. This influence can be expressed through use of the following variables:

$P_{S|M}(i,j)$ = Probability that a target moving to (i,j) will stop

$P_{\alpha|S}(i,j)$ = Probability that a target stopped at (i,j) will leave in direction α , and

$P_{\alpha|\alpha'}(i,j)$ = Probability that a target arriving at (i,j) from direction α' will leave at direction α .

(U) These transitional probabilities can be

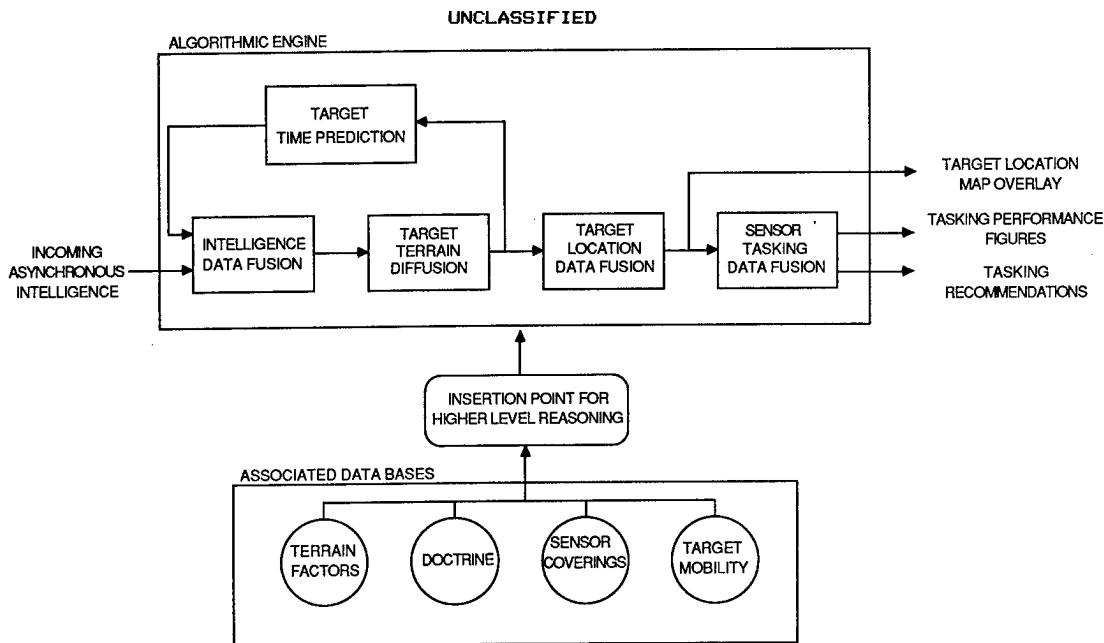


Figure 2. (U) System Overview

derived on the basis of deployment doctrain and terrain constraints as a function of time and stored in a data base, Figure 2. Alternatively, the development process can be mediated in real time through use of such higher level reasoning as Expert Systems analysis. Such an approach allows reasoning to be broken naturally between numeric and symbolic processors.

(U) Lacking higher reasoning or terrain constraints, the following deployment defaults are meaningful:

$P_{S|M}(i,j)$ = a value reflecting the general suitability of the terrain, and

$P_{\alpha|S}(i,j)$ = indicating independence of the new and old track directions.

(U) If the target is transversing open terrain, then its direction will likely be determined by the local terrain suitability. Thus its future directional probabilities can be given on homogeneous terrain as:

(U) $P_{\alpha|\alpha'}(i,j) = P_L[\text{Mod}_{360}(\text{Abs}(\alpha - \alpha'))]$ (1)
where,

$P_L(\beta)$ = the probability of a delta angle turn of β .

(U) If the target is moving on a road, then it will most likely stay on the road and not reverse its direction. Alternately, it could depart the road for open terrain at angle α with lesser probabilities. One can rather straightforwardly write an equation for this case similar to that of Eq. 1.

(U) Velocity

(U) Just as terrain and doctrine can exert influence on the target's directional tendencies, they also exert powerful influences on the target's velocity. Velocity is re-expressed here as that time required to leave one point on the grid and arrive at another:

$\eta_{\alpha}(i,j)$ = Time units required for a target leaving (i,j) at direction α to reach the next grid point.

This time variable should include a $\sqrt{2}$ adjustment for distance differences between adjacent points on the square grid.

(U) The variable, $\eta(i,j)$, is an excellent point where higher level reasoning processes can interact with and control the algorithmic time projection mechanisms. It is algorithmically straightforward to allow $\eta(i,j)$ to be a real time function dependent upon deployment considerations and doctrine.

(U) TARGET STATE PROJECTIONS

(U) Two forms of target state projections are required: spatial diffusion over terrain and velocity prediction over time.

(U) Terrain Diffusion

(U) At any moment the geometric state of the target is given by its location (i,j) , its orientation α , and whether it is stopped or moving. This leads to the following definitions:

$P_A(i,j,\alpha)$ = Probability of arriving at (i,j) from direction α at time t ,

$P_L(i,j,\alpha)$ = Probability of leaving (i,j) at direction α at time t,

$P_S(i,j)$ = Probability of being stopped at (i,j) at time t,

$P'_S(i,j)$ = Probability of being stopped at (i,j) at time t+1.

(U) The probability sum of a target being located or passing through a grid point is constant at each time instance. This results in the following equations for the probability of the target being stopped at grid point (i,j) at time t+1:

$$(U) P'_S(i,j) = (1 - \sum_{\alpha} P_{\alpha|S}(i,j)) P_S(i,j) + P_{S|M}(i,j) \sum_{\alpha} P_A(i,j,\alpha) \quad (2)$$

The probability of the target leaving point (i,j) at direction α at time t+1 is:

$$(U) P_L(i,j,\alpha) = P_{\alpha|S}(i,j) P_S(i,j) + \sum_{\alpha'} P_{\alpha|\alpha'}(i,j) P_A(i,j,\alpha') \quad (3)$$

These two equations must be executed at every grid point (i,j) at each time instance t.

(U) Prediction Over Time

(U) Target velocity effects have not been taken into account in the above terrain diffusion process. As the target leaves the grid point (i,j) at direction α , time $\eta_{\alpha}(i,j)$ will elapse before it can reach one of the eight neighboring grid points. This implies that the leaving probability, $P_L(i,j,\alpha)$, at time t will contribute to the arrival probability,

$$P_A(i',j',t+\eta_{\alpha}(i,j),360^\circ-\alpha).$$

(U) Observe that in accordance with laws of conservation, the total target probability over the grid must sum to one at each new time instance. Thus at the end of each time prediction instance and before the next terrain diffusion projection, the total grid probability must be renormalized to one.

(U) DATA FUSION

(U) As the target state projection continues, the probabilistic target location likelihoods will diffuse throughout the terrain grid. The target location will become totally lost unless new intelligence report updates are infused.

(U) At any time instance, the intelligence analyst may want to see a probabilistic plot of potential target locations. This involves a form of data fusion whereby the target state data is reduced to location probabilities.

(U) Finally, there will be recurring need to task available sensor resources. This

requires time sensitive fusion of sensor coverage characteristics with the corresponding target state data.

(U) Intelligence Updates

(U) Ideally, information would arrive continuously allowing the target location to be known at all times. Unfortunately, sensors must be tasked based on their next availability, and then the resulting data collection analyzed. This takes time and there is no assurance that the target intelligence collected will be of significant value. Meanwhile the target will be slipping away. As a result there is a high premium placed on the use of any and all arriving information, even if its arrival rate is irregular.

(U) Assume that at various times T_N , probabilistic information, $P_N(i,j)$, bearing upon the location of the target will be received. This new data will then need to be fused with the time prediction at T_N . This is most easily done by likelihood readjustment of the time prediction:

$$(U) P'_A(i,j,\alpha) = \frac{P_N(i,j) P_A(i,j,\alpha)}{L} \quad (4)$$

$$(U) P'_S(i,j) = \frac{P_N(i,j) P'_S(i,j)}{L} \quad (5)$$

where conditioning is on L:

$$(U) L = \sum_{i,j} P_N(i,j) [P_S(i,j) + \sum_{\alpha} P_A(i,j,\alpha)] \quad (6)$$

(U) Figure 2 shows the imposition of the data fusion update function into the target state projection cycle as an asynchronous modifier.

(U) Projected Target Locations

(U) The predicted location of the target on the grid at time t can be given as:

$$(U) P_P(i,j) = P'_S(i,j) + \sum_{\alpha} P'_A(i,j,\alpha) \quad (7)$$

(U) Sensor Tasking

(U) Since the purpose of sensing is to locate the target, it is reasonable to drive the sensor tasking directly from the target location prediction of Eq. 7. Considering imagery as the sensor of primary choice, at each time, t, it is most likely that the sensor can only examine a ground footprint, F:

$$(U) F(i,j) = 1 \quad \text{for all } (i,j) \text{ grid points within the sensor's field of view centered on grid point } (u,v), \\ = 0 \quad \text{elsewhere.} \quad (8)$$

(U) The size of F will usually be less than the non-zero region of probability of target location on the grid. Obviously it is then of interest to place this footprint at that

grid location, (u,v) , that maximizes the imagery sensor's probability of target detection, P_I . This can be done by the following action:

$$(U) (u,v) = \text{ArgMax}_{u,v} [P_I] \quad (9)$$

$$(U) (u,v) = \text{ArgMax}_{u,v} \left[\sum_{i,j} F(i-u, j-v) P_p(i,j) \right] \quad (10)$$

(U) RESULTS

(U) The above state projection and data fusion framework is presented in a mathematical and probabilistic framework intended to be tailored to meet the particular broad area search problem at hand. The following two case studies should be useful in understanding the general working of the framework.

(U) A Simplified Example

(U) Use of the algorithmic processes presented above is illustrated in graphical form in Figure 3.

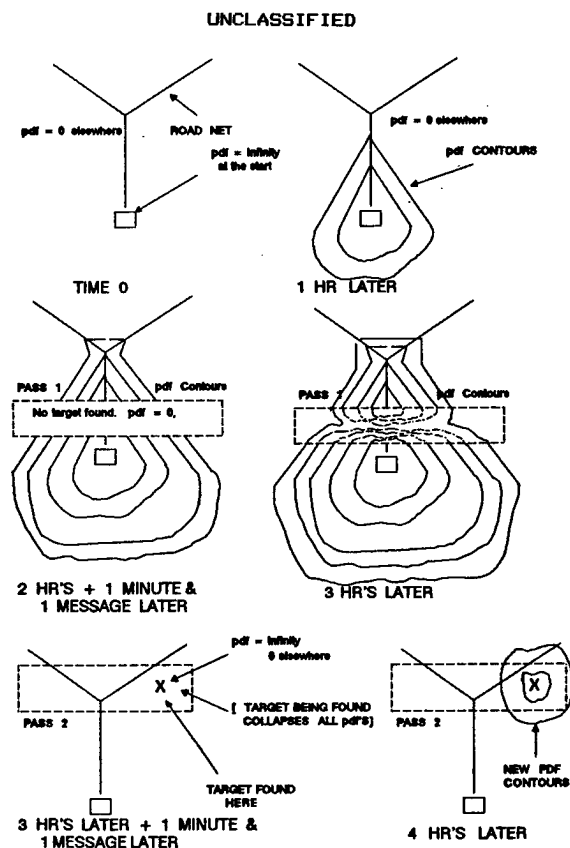


Figure 3. (U) A Graphical Example

(U) At time zero, a target vehicle is found at the end of a road; therefore, the location probability is set to one there and zero elsewhere. The vehicle is expected to move, so the location probabilities are projected forward one hour.

(U) The expected target location now lies within a teardrop shape with minor axes along open terrain and major axis along the road. The location probability projection is next moved forward two hours into the future. At this time a message is received that the target was not found in a sweep corridor. Meanwhile, the location probability contours have continued to expand in all directions and have begun to follow the branch of the road. Note that the location probability function has; however, been set to zero within the sweep corridor. This increases the probability of seeing the target elsewhere.

(U) At three hours, it is seen that the outer contour of the location probability function has expanded further and that the sweep corridor of one hour ago is starting to fill in.

(U) A second message is now received that the target has been found on the right branching road. Instantly, the location probability function collapses to a point probability of one with zero value elsewhere.

(U) The target location/data fusion process is never really completed since the target can continue to move. Thus as the location probability projection is continued forward to four hours, it is seen that the target location is again becoming probabilistic over a widening region.

(U) In this simple example, there was no attempt to task the sensor in an optimal manner. Rather, the data fusion process was limited to opportunistic collection. This lies within the true purpose of the techniques espoused above -- extensibility to various broad area search data fusion scenarios.

(U) An Expanded Simulation

(U) A larger simulation of the process of maintaining contact of relocatable targets over broad areas has also been developed. Terrain trafficability factors incorporated include primary and secondary road networks, terrain elevation, railroads, hydrographic and forestation features of Figure 4. Figure 4 also shows the initial probability function as a small dot at the target garrison. A rectangular footprint shows the latest imaging sensor check for activity.

(U) Results indicate target mobilization activity and the projection process is activated. Figures 5 and 6 show the broadening of location probability as a function of time. The brightness of the probability function is a direct indication of the strength of the location estimate. Observe that the resulting location

UNCLASSIFIED



Figure 4. (U) Initial Garrisoned Probability

probabilities mainly follow the road while being blocked by the presence of the river.

(U) As real time catches up with the projected time of Figure 6, opportunistic radio directional finding reports, ellipses in Figure 7, place the target generally away from the river. Fusion of these reports with the location function of Figure 5 results in the alteration of the location probabilities away from the river. Reprediction forward again to the projected time of Figure 6 now results in a modified re-expansion of the location probability, Figure 8.

UNCLASSIFIED



Figure 5. (U) First Time Projection Forward

UNCLASSIFIED



Figure 6. (U) Second Time Projection Forward

UNCLASSIFIED

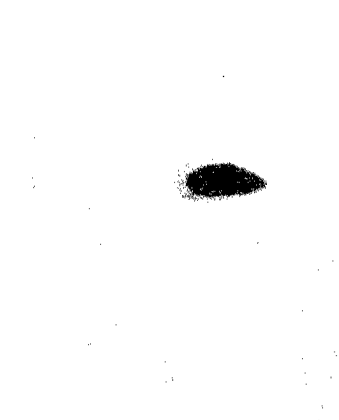


Figure 7. (U) Fusion of Intelligence Against First Time Projection

UNCLASSIFIED

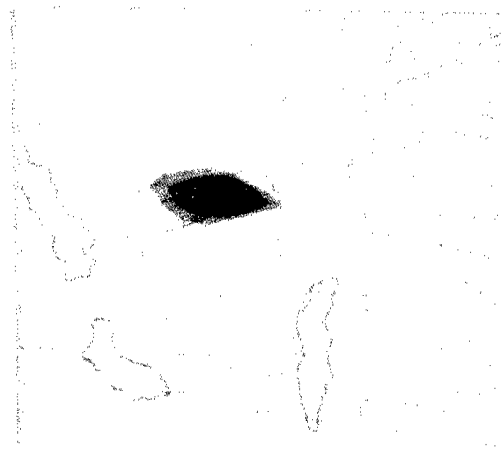


Figure 8. (U) Reprediction of Second Time, Sensor Tasking Recommendation

(U) Further time expansion will require increasingly large areas be searched. As a result it is decided that an imagery sensor pass be tasked before location becomes yet harder. The next available sensor type is identified to the algorithmic engine and it centers the footprint, shown in Figure 8, to the region of highest target acquisition probability. The analyst accepts the recommendation and the sensor is so tasked. Upon later collection analysis no target is found, the collection ground footprint is cleared of probability, and the remaining probability is increased. See Figure 9 where it is shown that the algorithmic engine makes use even of the negative information that the target is not present.

UNCLASSIFIED



Figure 9. (U) Fusion of Collection Analysis Against Second Time Projection

(U) The situation is now becoming more serious. Further time projection shows by the time of the next possible sensor retasking, the location probability will have significantly grown, Figure 10. Also, the prior collection footprint is rapidly filling in. Had there been pockets of uncertainty in the prior image collection analysis, then this filling would have been even more complete by this time. For instance, stands of trees will normally obscure many relatively small regions from sensor view. The footprint of Figure 10 would then have contained small pockets of non-zero probability which would have more rapidly filled in than before.

(U) Again a new image sensor tasking is requested. The algorithmic engine easily places the sensor footprint on the main lobe of the probability distribution, Figure 10. Since the current sensor collection footprint is well orientated against the main lobe the recommended tasking is accepted.

(U) Upon analysis of the collected image,

UNCLASSIFIED

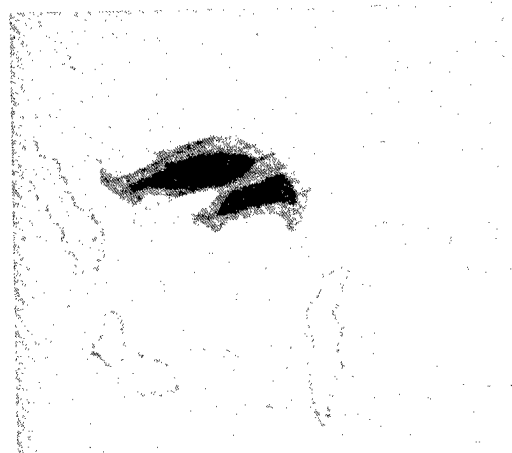


Figure 10. (U) Third Time Projection Forward and New Sensor Tasking

the target is found. This results in collapse of the location function to the target's location, Figure 11.

(U) Again, even though the target has been found, the situation can not remain stable as the target continues to move. Indeed, the later time projection of Figure 12 shows the target location uncertainty is again expanding in area.

UNCLASSIFIED



Figure 11. (U) The Target Is Reacquired!

(U) CONCLUSION

(U) In the kind of broad area search situation envisioned here, the problem is never ending, but the techniques outlined here can insure best use of all available resources.

(U) An algorithmic engine combining target projection and data fusion functions can almost certainly keep better track of the intricacies of the broad area search problem

UNCLASSIFIED

Figure 12. (U) The Process Continues...

than unaided human initiative. The overall process should be used synergetically; however, with the analyst retaining full control. Various projections can be made into the future; relative confidences can be assigned to projections and to new incoming intelligence; the results can be refused; sensor resources selected and reselected; and alternate sensor footprint optimizations remade as desired. The framework is available.

AN ESM DATA FUSION PROCESS FOR TRACKING MASS RAIDS

Thomas M. Hart

**The MITRE Corporation
Bedford, Massachusetts 01730
(617) 271-8503**

Mass raid attack scenarios are considered viable in some regions of the world. In these scenarios, thousands of hostile aircraft transit narrow attack corridors rapidly and are accompanied by severe radar electronic countermeasures (ECM). Maintaining an adequate air picture based on conventional active radar tracking of individual aircraft or small formations may not be possible in this type of environment.

An automated mass raid tracking technique has been developed which uses data from cooperating airborne Electronic Support Measures (ESM) systems to restore the air picture which might be lost to radars in a severe ECM environment. The technique exploits the kinematic and electronic signatures of mass raids to detect and track raid leading edges and centers of mass and to locate raid corridors.

In this technique, emitters of hostile aircraft which are expected to be well represented in the leading edge and center of mass components of a mass raid attack are targeted with ESM. Detections on emitters associated with the leading edge components of the raid are bundled to form cluster reports which represent angular regions of significant emitter activity. The location and extent of the leading edge component are computed by triangulation with similar information told in from a cooperating remote ESM system via tactical digital data link. The location and extent of the centers of mass are determined in a similar fashion and an air picture is constructed by combining the results of the two processes. Leading edge history trails define the raid corridors over time. Ghosting, although possible, is significantly reduced and responds well to logics which discriminate on kinematic criteria and geographical constraints.

Algorithms for this technique have been developed and implemented in a simulation of an ESM system in a large-scale mass raid scenario. The paper describes the algorithm and presents results of the simulation.

AN ESM DATA FUSION PROCESS FOR TRACKING MASS RAIDS

Thomas M. Hart

The MITRE Corporation
Bedford, Massachusetts 01730
(617) 271-8503

INTRODUCTION

Mass raid attack scenarios which are characterized by high target densities and severe electronic countermeasures (ECM) may constitute an environment which would seriously impair surveillance systems designed to provide active radar tracks on individual targets. Although major radar electronic counter countermeasures improvements may be an important part of a long term solution, there are opportunities to address the problem sooner.

Methods for building an air picture which employ information from other surveillance sensors is one alternative. Any such methods should consider the problems introduced by the large number of targets in the environment and the proximity of those targets to each other. Mass raid attacks represent extreme examples of closely spaced multiple targets. Such an environment places enormous demands on any tracker.

This paper outlines a method for building an air picture in a mass raid environment. This method is called Strobe Bundling. It uses information potentially available from passive sensors such as Electronic Support Measures (ESM). It employs clustering and triangulation techniques to locate raid leading edges, centers of mass and raid corridors. It is intended to provide display information useful for broadcast air control of tactical resources. It does not attempt to maintain individual tracks on targets.

This paper motivates the Strobe Bundling method by discussing a mass raid environment and the sensor information which may be available in it. A description of the method is given as well as simulation results. The report concludes with a discussion of suggestions for further development.

BACKGROUND

Mass raid scenarios can be characterized by the example shown in figure 1. In such scenarios, many hundreds of hostile aircraft transit narrow attack corridors. In addition, a large amount of radar ECM accompanies the attacking aircraft. This ECM may take various forms; ground-based and stand-off jamming, escort and self-screen jamming.

Mass raids are composed of aircraft with different specific missions. Typically, these missions include bombing, fighter escort, jamming, chaff laying, ground attack, and reconnaissance. The nature of these missions lends a certain structure to the raid.

Reconnaissance aircraft fly at high speeds and elevations, detached from the main bulk of the raid. Chaff layers precede the bombing and attack aircraft and define a leading edge to the raid components in each attack corridor. Bombers and their fighter escorts comprise the bulk of the raid in each corridor and indicate the raid center of mass.

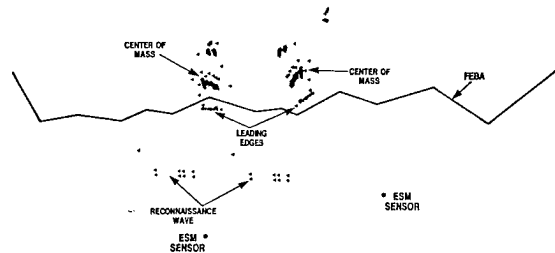


Figure 1. MASS RAID SCENARIO

The concentration of hostile forces into narrow transit corridors is an attempt to break through front-line air defenses by overwhelming numerical superiority. The extensive jamming employed reduces radar effectiveness to a point where tracking of individual targets cannot be adequately performed to provide the surveillance information necessary to weapons control and other C² functions.

The loss of adequate radar data does not preclude the existence of other usable sensor inputs. In particular, electromagnetic emissions from on-board avionics equipment can be targeted by passive sensors such as ESM. Although ESM detections do not provide range estimates, they can be used to estimate the direction to an emitter. Further, ESM systems have the capability to measure parameters of detected signals. These parameters may include frequency, pulse width, pulse repetition frequency, and modulation characteristics. Such parametric information from a detected emitter can then be compared to emitter library entries, and an assessment made of the emitter type made. Emitter type estimates can be used to make an estimate of platform identification.

Although distributed ESM sensors can be used to perform cooperative-passive tracking, this surveillance method requires sufficient system sensitivity to assure that cooperating sensors detect the same emitters. Moreover, the large number of emitter

reports potentially available in a mass raid environment can create a large number of false intersections or ghosts. Effective deghosting requires that all the reports from cooperating sensors be available for processing. This processing can be done at one of the sensors or it can be done at a central processing location. Regardless of where it is done, a large data transfer capability would be needed to support the sharing of information on individual reports.

The approach taken to mitigate these requirements is to provide *raid* information rather than *target* information. Raid information would include such characteristics as the location of the leading edge and center-of-mass of the raids, physical extent and target count, and estimates of the movement of the raids through the corridors. Processing of individual ESM reports is greatly reduced; most processing occurs on combinations of data from individual reports.

ESM SENSOR MODEL

The ESM system modeled in this study is a general purpose, wideband passive surveillance sensor for determining direction-of-arrival (DOA) to a target, as well as emitter and platform information on the targeted signal. Parametric measurements of a signal can be used to distinguish between two emitters even if the emitters are not distinguishable in DOA.

The ESM model used for this study assumes that emitter reports along the same DOA azimuth are resolvable (by means of parametric data comparison) and that the emitter class of each report can be determined. An emitter class is composed of emitters associated with aircraft performing similar missions.

STROBE BUNDLING ALGORITHM

The Strobe Bundling algorithm collects ESM reports from individual emitters into histograms and separates these reports into clusters of significant report density. Thus, data is twice fused to reduce subsequent processing: once when individual reports are placed in histogram bins, and again when raid clusters are formed from the histogram.

Raid clusters from cooperating ESM sensors are then triangulated in a manner similar to conventional triangulation of individual reports. By triangulating clusters associated with different emitter classes, leading edge and center of mass estimates of the raid are obtained. Since many individual reports are replaced with very few cluster reports, the ghosting problem is significantly less than would be encountered using individual reports.

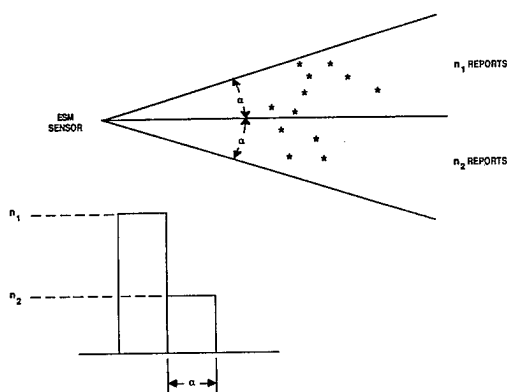


Figure 2. STROBE BUNDLING

ESM reports from individual targets close in azimuth are collected into angular bins. Once binned, there is no requirement to keep track of individual reports and so individual report DOA

information is replaced by a frequency count for each bin. A histogram is produced of DOA versus number of ESM detections per bin. Figure 2 is a stylized illustration of the binning process. Each bin in the histogram has angular width α . The height of each bin corresponds to the number of ESM detections whose measured DOAs are within the limits of that bin.

An example histogram is shown in figure 3. The X-axis is DOA to scenario aircraft from the ESM sensor. DOA is referenced to True North. The Y-axis is number of emitters detected.

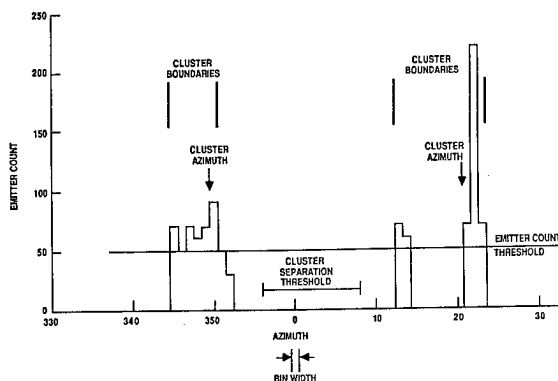


Figure 3. EMITTER COUNT HISTOGRAM

Clusters of significant target density are formed from the histogram. A cluster report consists of a cluster azimuth and cluster boundaries. The cluster azimuth is a single numerical value which represents an average DOA for the cluster. Cluster boundaries indicate the angular extent of the cluster. A cluster report can be thought of as a "thickened" strobe; one report representing a potentially large number of ESM reports.

Calculation of cluster azimuths consists of four functions:

- "Noise" filtering
- Cluster separation
- Cluster separation determination
- Cluster azimuth calculation

Each of these functions will be discussed below.

"Noise" Filtering

The emitter count for each bin in the histogram is compared to a threshold. This thresholding distinguishes low emitter count, noisy bins from bins containing the significant portions of the raids. The threshold value used is a function of the largest emitter count found in any bin across the angular field of view and an empirically determined constant. Thus

$$\text{THRESH1} = \text{NOISECON} \cdot \max_i \{f_i\} \quad (1)$$

where

THRESH1 = emitter count threshold

NOISECON = constant

f_i = emitter count for bin i .

Bin j is a "zero" bin if $f_j < \text{THRESH1}$. If the emitter count for a bin exceeds THRESH1 it is called a "non-zero" bin. Zero and non-zero bin designations are used for cluster separation processing.

Cluster Separation

Cluster separation is determined by comparing the number of consecutive zero bins separating two non-zero bins to a second threshold, THRESH2. Thus, if bins j and $j + k + 1$ are non-zero bins, and bins $j + 1, j + 2, \dots, j + k$ are all zero bins, bins j and $j + k + 1$ are considered in separate clusters if $k \geq \text{THRESH2}$.

This second threshold is determined by:

$$\text{THRESH2} = \text{SEP}/\alpha \quad (2)$$

α = bin width measured in radians

SEP = minimum separation distance between raid clusters.

The value of SEP reflects a judgment as to what configuration of targets in a given scenario represent different raid components.

Cluster Boundary Determination

Cluster boundaries indicate the angular extent of the raid components.

They are calculated from the periphery bins by:

$$\text{CLUSTBNDRY} = \alpha \cdot j \quad (3)$$

where

CLUSTBNDRY = cluster boundary value

j = histogram bin number of a periphery bin.

Cluster Azimuth Calculation

The cluster azimuth is calculated as the average angular value of the bins in a cluster weighted by the emitter counts of the bins in that cluster:

$$\text{CLUSTAZ} = \alpha \left[\text{NINT} \left[\frac{\sum_j \cdot f_j}{\sum f_j} \right] - 1/2 \right] \cdot \text{Mod} \frac{2\pi}{\alpha} \quad (4)$$

where

CLUSTAZ = cluster azimuth for a given cluster

f_j = emitter count in bin j .

NINT is the "nearest integer" operator. The nearest integer operation locates the center azimuth bin. The sums in the numerator and denominator of the operator argument range over the histogram bin numbers associated with the cluster. Note that the $-1/2$ term positions the cluster azimuth in the middle of the appropriate center azimuth bin. The calculation is modulo the number of bins to account for clusters which straddle the discontinuity in the bin labelling scheme that occurs at True North.

COOPERATIVE PROCESSING

Cluster reports from a single ESM sensor provide raid detection information and an estimate of direction to the major raid components. Azimuthal motion can be estimated from cluster azimuths calculated at different times.

Single sensor cluster reports suffer however from the same restriction as individual ESM reports in that they do not provide range information. Range estimates can be obtained using triangulation

if cluster reports are available from a cooperating remote ESM sensor via a tactical digital communications link. Triangulation of cluster azimuth values estimate the raid centroids. Triangulation of cluster boundaries estimate spatial extent of the raid clusters.

It is important to note that the amount of ESM data required to be transmitted over the communications link is drastically reduced when cluster reports (or at most, histogram information) instead of individual ESM report data is sent. In addition, the number of intersections created by triangulating cluster reports is generally a small fraction of those that would be created while doing conventional cooperative ranging with individual reports. The number of ghosts drops approximately by the difference of the squares of the number of individual reports and the number of cluster reports.

Raid structure can be exploited by the histogram/cluster analysis algorithm. Aircraft with different missions require different classes of emitters. Modern ESM sensors can be expected to distinguish and sort signals from different classes. Histograms can be constructed for each set of class-sorted signals. The cluster reports created from each of these histograms can then be combined to provide more information on raid structure than if emitter signals are binned without regard to class.

SIMULATION RESULTS

The algorithm was tested by simulation in the scenarios described earlier. Figure 4 shows combined results when each ESM sensor processes emitter reports from raid leading edge and center of mass signature aircraft separately. Cluster boundaries are not shown. Instead, the intersection of cluster boundaries from cooperating ESM sensors are shown. The quadrangular sets are called raid extent boxes. Cluster azimuth strobes from both ESM sensors within the raid extent boxes are also shown.

Figure 5 illustrates the set of leading edge raid extent boxes over several processing updates. Note that the raid extent boxes effectively trace out the raid corridors.

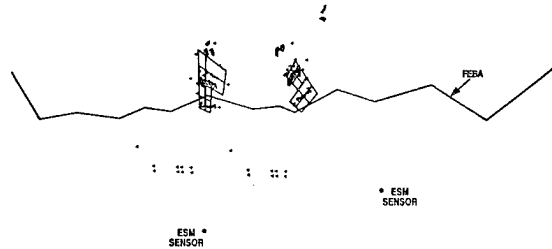


Figure 4. LEADING EDGE AND CENTER OF MASS CLUSTER REPORTS

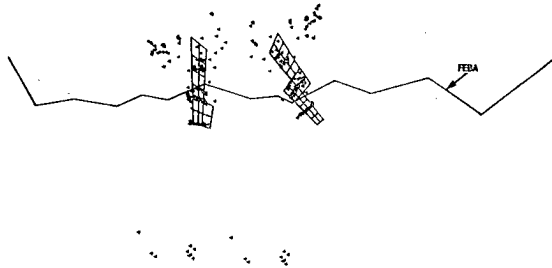


Figure 5. RAID CORRIDOR DETECTION

CONCLUSION

The current implementation of the Strobe Bundling algorithm represents a preliminary effort to address the mass raid surveillance problem using distributed passive sensors. Further refinement of the design would concentrate on making the algorithm more robust. For example, the logic involved in cluster separation uses constant value parameters: the emitter count threshold constant NOISECON, the separation threshold THRESH2, etc. These parameters could be allowed to vary to reflect changes in the environment. Similarly, the present algorithm bins the data using an empirically determined constant bin width. This constant value is the best value on the average, but certainly is more useful in some geometries rather than others. A more robust design would be to bin the data simultaneously into two histograms, one with a relatively wide bin width, and one with a relatively narrow bin width. The wide bins should more clearly indicate clusters: emitter count distributions will be more pronounced. On the other hand, finer grain estimates of cluster azimuth values are possible with smaller bin widths. Results

from both histograms can then be combined in a complementary way.

At present, all processing is done with just one cycle time worth of data. Thus, prior results are not considered when doing current processing. This memory-less design can result in inconsistent results from update to update. Here, "inconsistent" refers to the number of clusters determined, the sizes of the raid extent boxes, etc. A feedback mechanism which checks for consistency with past results would improve this situation.

A major area of further investigation is in determining a measure of the algorithm's performance. This effort is complicated by the lack of how to identify features of the raid which the algorithm measures, and then to translate these features into numerical values. The definition of measures of effectiveness of the algorithm is crucial, so that results can be compared, and improvements identified.

THIS PAGE LEFT BLANK INTENTIONALLY

**AN APPLICATION OF COMPUTERIZED AXIAL TOMOGRAPHY (CAT)
TECHNOLOGY TO MASS RAID TRACKING**

John K. Barr

**The MITRE Corporation
Bedford, Massachusetts 01730**

ABSTRACT

Mass raid attack scenarios are considered viable in some regions of the world. In these scenarios, thousands of hostile aircraft transit narrow attack corridors rapidly and are accompanied by severe ECM. Maintaining an adequate air picture based on conventional active radar tracking of individual aircraft or small formations may not be possible in this type of environment.

Density reconstruction is a technique for combining electronic support measures (ESM) detection data on a mass raid which has been collected at three or more cooperating ESM platforms to reconstruct the size and shape of the raid. The data are processed to construct a digital representation of the mass raid called a target density map.

Density reconstruction is based on computerized axial tomography (CAT) technology. In medical applications, CAT scanners measure the summed physical density of an object, such as a human chest cavity, from several viewing angles to reconstruct the size and shape of the object. In density reconstruction, ESM detection data collected from several viewing angles are processed to measure the summed emitter density of a mass raid and reconstruct its size and shape.

Algorithms for this technique have been developed and implemented in a simulation of an ESM system in a large-scale mass raid scenario. The paper describes the density reconstruction algorithms and presents the results of the simulation.

AN APPLICATION OF COMPUTERIZED AXIAL TOMOGRAPHY (CAT)
TECHNOLOGY TO MASS RAID TRACKING

John K. Barr

The MITRE Corporation
Bedford, Massachusetts 01730

INTRODUCTION

In this paper we discuss a new technique, target density reconstruction, for using passive angle-only data from multiple sensor sites to derive the location of large numbers of airborne targets. In a high target density, intense jamming environment, the tracking of individual targets by radar may be severely degraded, and passive sensors such as electronic support measures (ESM) may provide the only target data. Cooperative passive tracking using ESM data may itself be difficult when many targets are present. The target density reconstruction method described here is designed to gain the maximum possible raid location information from passive data. Instead of individual target tracks, it generates a target density map (TDM) with estimates of the number of targets in each of many small grid cells covering the surveillance area. Internally, target density reconstruction is similar to a computerized axial tomography (CAT) scanner.

RATIONALE

Target density reconstruction is a means of deriving the maximum amount of summary raid location information when standard tracking from radar and ESM has been degraded and ESM cooperative tracking is not possible.

The environment for use of target density reconstruction is a massive air attack with hundreds of aircraft in multiple waves crossing a border in one or more corridors. The invading aircraft are protected by jamming of the defending radars. Figure 1 shows the locations of aircraft several minutes into a generic raid of this type. The dense raid environment is described in detail in Ref. 1. The intense jamming during the raid could seriously degrade the ability to radar track individual aircraft.

We assume that a netted system of ESM sensors is in place, such that any potential aircraft location is in line of sight to at least three ESM sites. Each ESM system can detect and measure the radar and other signals being emitted by the aircraft in its field of view. The azimuth angle of arrival (AOA) to each target at each ESM site is determined, along with signal parameters such as radio frequency (RF) and pulse repetition interval (PRI).

With a small number of targets it is generally possible to combine the AOAs at two ESM sites to triangulate a target, and to

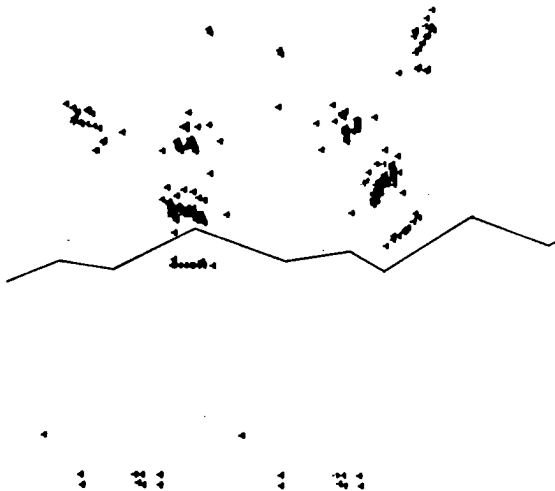


Figure 1. MASS RAID TARGET LOCATIONS

perform cooperative passive tracking. However, this form of tracking requires correct association of the angle strobe at one ESM site with the strobe generated by the same emitter target at the other ESM site. With N targets to be tracked, N strobes will be generated at each of two ESM sites, leading to N^2 strobe crossings. Each crossing is a possible target location, but only N of them are correct. The remaining $N^2 - N$ crossings are called "ghosts." With data from three sites, the ghosting problem can be even worse, with on the order of N^3 crossings generated. Figure 2 shows the strobe pattern produced at three passive sensor sites by the dense raid shown in figure 1.

In less dense target situations, the ESM signal measurements can be used to sort out the valid strobe pairings. But with the large number of targets being postulated here, signals from different emitters may overlap in their measured parameters, becoming indistinguishable to the ESM system. Emitters may also change their operating characteristics between the time of detection at one ESM site to the time of detection at another, making cross-correlation difficult or impossible. For example, the ESM system may detect a signal once each 10 or more seconds, while the target emitter is

changing RF and/or PRI many times a second. For these reasons, cooperative tracking by ESM sites may be degraded in a dense raid environment.

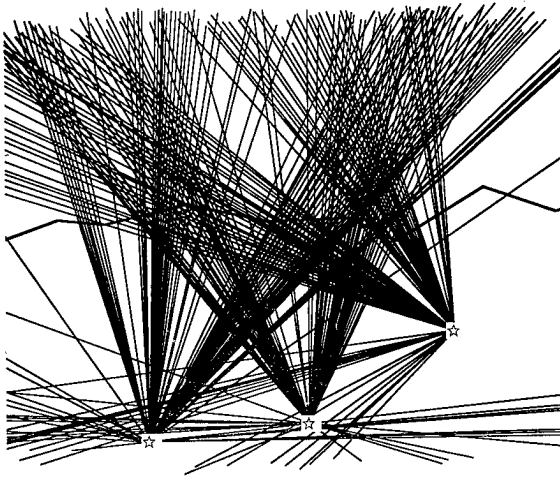


Figure 2. STROBES AT THREE ESM SITES

The method of target density reconstruction is intended to fill the gap left by the missing radar and cooperative passive trackers, by providing an estimate of the general raid situation in as much detail as possible. Although individual tracks are not derived, the target density map gives an estimate of the number of targets in each 10-km by 10-km grid cell. Figure 3 shows a TDM generated directly from the true target locations given in figure 1. The goal of an estimated TDM using ESM measurement data would be to come as close as possible to figure 3. The accuracy of the estimated TDM depends on the number of ESM sites measuring the grid cell and also on the accuracy of the ESM system's measurements of AOA and signal parameters. The method makes the fullest possible use of the available input data.

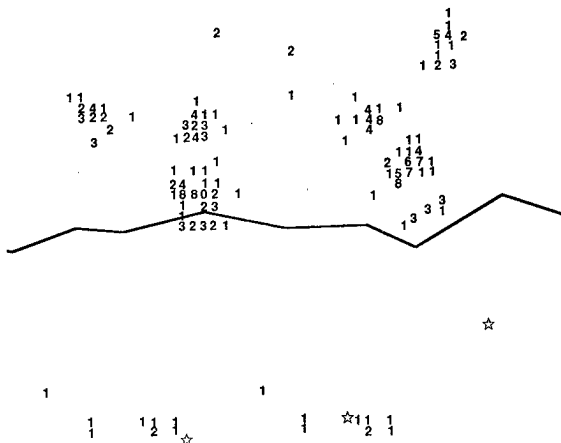


Figure 3. TDM TRUE DENSITIES

TARGET DENSITY RECONSTRUCTION PROCESSING

Sorting of Signals into Subgroups

The use of ESM data for target location depends critically upon the ability of the ESM system to discriminate between emitters based on signal characteristics. When signal features are measured and data from different ESM sites are combined, reports with equal signal parameter values can be processed as a separate

group. This breaks up one large data association problem into many smaller such problems, each of which can be more easily solved. The ideal, of course, would be to have only one ESM report per site in a group, which allows for cooperative tracking. If the number of targets in the group is small enough, relative to the angular resolution of the ESM, discrete tests can eliminate many ghost intersections. The principal test rejects a strobe intersection from two ESM sites which is not corroborated by a strobe from a third site. Other more complicated tests can be employed, but are not discussed in this paper.

Density Reconstruction

Target density reconstruction is an application to ESM data of the general density reconstruction technique which has been successfully applied in such fields as radio astronomy and computerized axial tomography (CAT) scanning (see Refs. 2 and 3). The technique is basically a way of recreating the distribution of the contents of a region when all that is known about the region are the sums of that content along lines through it. There are two essential input requirements which must be satisfied for density reconstruction to work in the target problem. The first, which can be satisfied by ESM systems, but not by some other passive systems, is that the number of emitters be counted along each angle of arrival. The second requirement is that ESM data be available from at least three sites. The count data from only one site are not enough to perform any reconstruction, and two-site reconstruction is not generally useful. Once data from three or more sites are combined, however, density reconstruction can give a good picture of emitter location densities.

We now describe the computation method and apply it to the case of a dense raid scenario with ESM data. We also point out features of the dense raid/ESM problem which make ESM density reconstruction different from that in a CAT scanner or other typical applications. We will concentrate on the case of three sites. Processing is similar when more sites are present.

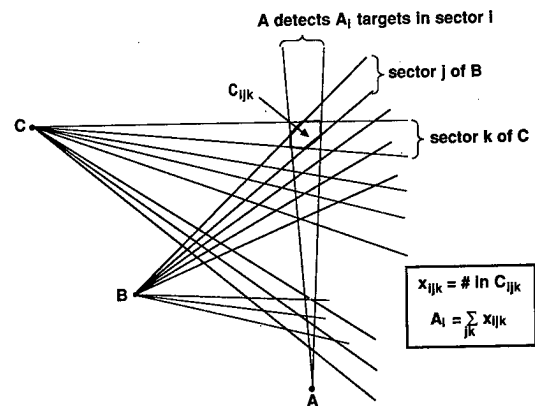


Figure 4. SECTOR EQUATION

In the ESM density reconstruction problem the inputs are the number of emitters detected in each angular sector, for all the cooperating sites. Examine, for example, the i th sector with respect to site A (see figure 4). The borders of sectors centered on other sites cut across sector i and divide it into many small odd-shaped pieces C_{ijk} , called calculation cells. Here, C_{ijk} is the calculation cell cut out by sector i from A, sector j from B, and sector k from C. If X_{ijk} is the number of targets in the calculation cell C_{ijk} and site A has measured A_i targets in sector i then

$$A_i = \sum_{(j,k) \in S_{A_i}} X_{ijk} \quad (1)$$

In the equation, S_{Ai} , is the set of pairs (j,k) such that C_{ijk} is a calculation cell in sector i of site A. Each sector defines one such equation relating calculation cell target counts to an angle sector count. The technique of density reconstruction is to solve for the X_{ijk} using all sector equations (Eq. 1) for all sites. If there are three sites and n angle sectors about each site, this means that 3n equations are defined. Only 3n-2 of the equations are independent, and there are in general many more than 3n-2 calculation cell unknowns X_{ijk} , so the system of equations Eq. 1 is badly underdetermined.

Three steps are used in ESM density reconstruction to reduce the size of the problem and find a reasonable solution. The first is to notice that, being a count of targets, no X_{ijk} could be negative. The X_{ijk} must then not only satisfy Eq. 1 but also

$$X_{ijk} \geq 0 \quad \text{for all } X_{ijk} \quad (2)$$

This step reduces the number of possible solutions, but does not provide a unique one.

The second and critical step uses the fact that many angle sectors contain no emitters at all, even in a dense raid scenario. If a platform emitter count is zero in a sector, then all the computation cells in the sector must be empty of emitters, and their X_{ijk} must be zero. These X_{ijk} and the sector equation can now be eliminated from further processing. Since many X_{ijk} are eliminated and only one equation is lost, this step improves the balance of equations against unknowns as well as reduces the computational load. If the ESM system has been able to divide the strobe data into several smaller data sets, then the improvement from step two is even sharper than when all strobes are processed together. This is because fewer targets and AOA reports allow for additional elimination of computation cells.

Even after steps one and two the system defined by Eq. 1 and Eq. 2 is still badly underdetermined, with no unique solution. The problem is then to find the best solution from among many possible ones, on the basis that even if that solution does not exactly match the true emitter locations it will be close enough to be useful. Two criteria have been proposed frequently as ways to select a best solution. Each algorithm is implemented in the real number system, with results rounded later for display. The first, called the algebraic reconstruction technique (ART), is an iterative quadratic optimization which converges to the smoothest solution, i.e., the one which minimizes

$$\sum_{ijk} (X_{ijk} - \bar{X})^2 \quad (3)$$

where \bar{X} is the average value over all the X_{ijk} . See Ref. 3 for details on ART. An update generated by ART does not automatically satisfy the non-negativity constraint (Eq. 2) on the solution. This constraint can be explicitly coded into the iteration by setting to zero those X_{ijk} which would otherwise be set to negative values. ART was the algorithm used in the first commercial CAT scanner, but has given consistently poor results in ESM and in the study (Ref. 4), where three-view reconstruction methods were compared.

The second criterion for choosing a best density solution from the set of possible solutions used a probabilistic approach to choose a maximum likelihood solution. The reasoning of Jaynes in Ref. 5 is applied to the target problem as follows.

Suppose there are N calculation cells and a total of R aircraft. We make the simplifying assumption that any "configuration" or specific placement of aircraft into the cells is equally probable. This assumption corresponds to not knowing the prior probabilities. There are N^R configurations, and each configuration

generates a particular density solution $\{X_{ijk}\}$, where, as above, X_{ijk} is the count in cell C_{ijk} . The number of configurations having the same density $\{X_{ijk}\}$ is the multiplicity

$$W = \frac{R!}{\prod_{ijk} (X_{ijk}!)} \quad (4)$$

Since we assume that each configuration is equally probable, the likelihood of a given density solution $\{X_{ijk}\}$ can be directly calculated, since it is just the multiplicity W of configurations having that density divided by the total number N^R of configurations, i.e.,

$$\text{Prob}(\{X_{ijk}\}) = W/N^R \quad (5)$$

The target density algorithm then seeks the maximum likelihood density solution over all densities which satisfy the ESM measurement constraint equations. As usual in maximum likelihood estimation, it is convenient to solve for the maximum logarithmic likelihood solution. Since in a given reconstruction the numbers N and R are fixed and known, the maximum likelihood solution occurs at the maximum of $\ln(W)$. In reconstruction, the number R of aircraft is large, allowing use of Stirling's approximation for $R!$, to give

$$\ln(W)/R \approx - \sum_{ijk} (X_{ijk}/R) \ln (X_{ijk}/R) \quad (6)$$

Since the sum on the right is the entropy $H(\{X_{ijk}\})$ of the vector of numbers $\{X_{ijk}\}$, maximizing $\ln(W)$ is equivalent to the well-known maximal entropy solution to the ESM angle sector equations. With K measurement constraint equations, the distribution of configurations near the maximal entropy solution is shown by the entropy concentration theorem (Ref. 5) to be chi-squared with $k = N - K - 1$ degrees of freedom. Thus for large R, the fraction F of configurations having entropy within ΔH of the peak value is given by

$$2R\Delta H = \chi_k^2 (1 - F) \quad (7)$$

Since we assume that each configuration has the same probability, this expression then can give an estimate of their distribution near the maximal entropy peak.

The maximal entropy solution is easily found by using the MENT iteration algorithm which converges to the unique entropy peak. It can be shown (Ref. 6) that the X_{ijk} in a maximal entropy solution have the form of a product

$$X_{ijk} = E_i E_j E_k \quad (8)$$

Then each sector equation can be written

$$A_i = \sum_{(j,k) \in S_{Ai}} X_{ijk} = \sum_{(j,k) \in S_{Ai}} E_i E_j E_k = E_i \sum_{(j,k) \in S_{Ai}} E_j E_k \quad (9)$$

and a solution must satisfy

$$E_i = \frac{A_i}{\sum_{(j,k) \in S_{Ai}} E_j E_k} \quad (10)$$

In the MENT iteration algorithm, each equation (Eq. 10) is used in turn to update the value of a single E_i from the current values of the E_j and E_k . The unknowns E_i , E_j , and E_k are initialized to 1. After three to five cycles through the entire set of sector equations, the E 's are changing by small increments, and the iterations are ended. The output X_{ijk} are then generated from Eq. 8. With 185 targets, the MENT algorithm required from 10 to 20 seconds to reconstruct target TDMs on a VAX 11/750 system.

The density reconstruction algorithm can be summarized as a setup stage, in which a minimal number of unknowns is found, and a second phase in which the linear equations relating the unknowns to the data are iteratively solved. Since the variables X_{ijk} refer to odd-shaped calculation cells in the surveillance region, some of them quite small, the resulting solution must be translated to a regular grid before being given to the user. This is done by adding together the densities of the calculation cells contained in a grid box and integerizing the result for output. If a cell has parts in different grid boxes, its calculated density is prorated among the boxes according to area.

Implementation of Density Reconstruction

The two main parts of target density reconstruction are the definition of the unknown densities to be calculated and the iterative solution for them. We now examine the different approaches which can be taken to implement the density equations and unknowns and the reasoning which has led to the present algorithm.

The most difficult part of density reconstruction is the definition of the variables X_{ijk} and their equations. This step is quite simple in CAT scanners and other applications of the technique, where a fixed 2-dimensional grid is used to define the areas whose densities must be calculated. In ESM density reconstruction such grids have given poor results, leading to the use instead of calculation cells described above.

The problem with using a grid to define the unknowns in ESM density reconstruction is that it cannot accurately represent the reconstruction geometry near the ESM sites. For example, if a sector angle size of 1 degree is used, then a 1-km grid size is adequate for areas further than 57 km from any site. A grid cell closer than this will not be contained in a single sector centered at the site so its variable X must be allocated to more than one sector equation, leading to inconsistent equations. To overcome this sampling problem, variable size grids were tried, but these were found to be very computationally expensive, as well as producing large numbers of variables.

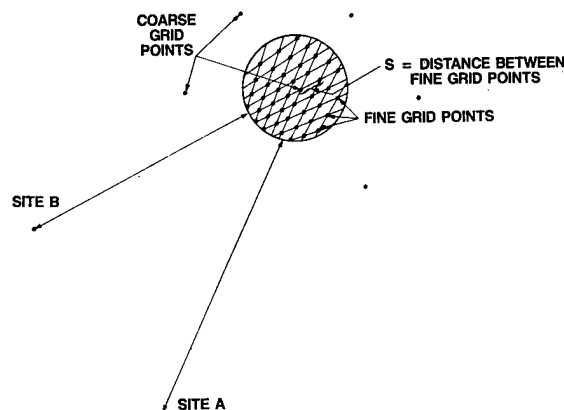


Figure 5. SAMPLING IN DENSITY RECONSTRUCTION

The method which we used replaces square grids with more general angle-sector sampling. In the grid definition processing, a coarse grid of equilateral triangles 50 km on a side covering the surveillance area except at points near the ESM sites is scanned. For each coarse grid point X , sectors from all platforms intersecting the region near X are checked. If none of these sectors has a positive emitter count, then the region is empty and is dropped from further processing. Otherwise, a fine grid is set up near X in the following way. Referring to figure 5, suppose that Site B is closest to X . Then the midlines of angle sectors from Site B through the area define one direction of the fine sampling. Sample points are placed on the midlines so as to give the coarsest possible sample spacing which is still adequate to represent all sector equations. Each point of the finer grid is checked for consistency between the three ESM sites: if any platform has detected no emitters in the sector containing the point then the point is rejected. Otherwise, it defines a new variable in the equations.

TARGET DENSITY MAP EXAMPLES

In this section we look at two TDMs generated by the target density reconstruction algorithm. Each TDM is an estimate using the same target locations as figure 1, so the ideal result would be a TDM as close as possible to figure 3. Angle measurement error was not modeled in this simulation, and an angle sector size of one degree was used. The stars in the figures show the locations of the three ESM sites.

Figure 6 shows a TDM generated on the assumption that the input ESM signal parameter data did not allow the signals to be sorted into subgroups for separate processing. This corresponds to the most difficult conditions for use of the density reconstruction algorithm. The figure shows that principal target groupings have been successfully reconstructed, although finer detail is missing, especially at points far from the ESM sites.

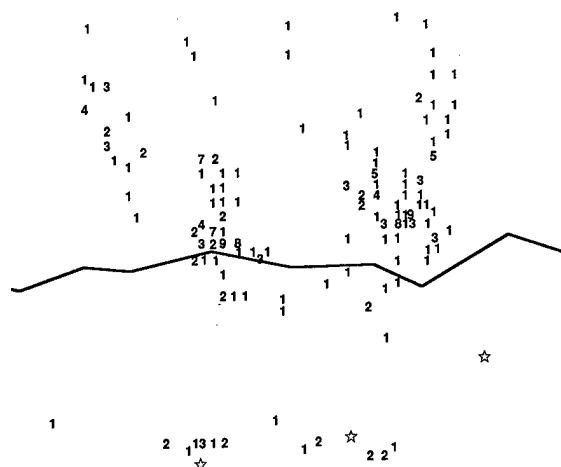


Figure 6. TDM: RECONSTRUCTED DENSITIES

Figure 7 is a TDM generated on the more favorable assumption that the ESM systems could divide the signals into 9 subgroups for processing. With groups of up to 20 emitters, the density reconstruction algorithm was able to give a detailed picture of raid groupings. The TDM also accurately shows isolated small groupings of targets. As an area increases in size (e.g., 50 km by 50 km), the total estimated target count there gets closer to the true value. However, the estimated counts in an individual box can differ significantly from the corresponding true value as given in figure 3.

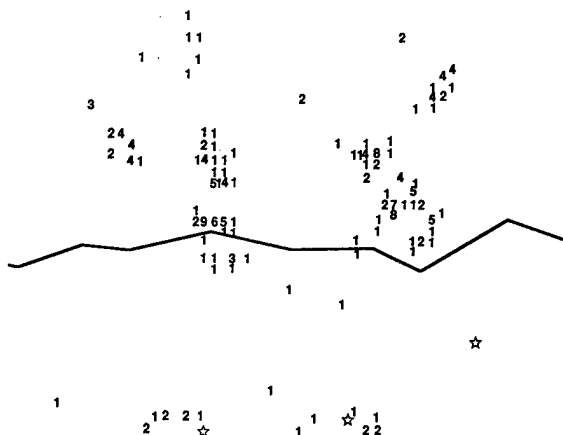


Figure 7. TDM: RECONSTRUCTED DENSITIES (NINE GROUPS)

CONCLUSION

We have
count data whi
ings. The infor
method and dis
possible use of

The proces
on that in a CAT
target problem. T

and the MENT algorithm to select a best TDM solution from the set of possible solutions.

Target density reconstruction could keep the user informed of major concentrations of air forces, and is intended as a backup in case primary radar is degraded, and if other sources such as ESM cooperative passive tracking are not available.

REFERENCES

1. T. Hart, "An ESM Data Fusion Process for Tracking Mass Raids," *Proceedings, Second Annual Tri-Service Data Fusion Symposium* (1988).
2. Y. Censor, "Finite Series-Expansion Reconstruction Methods," *Proc. IEEE*, Vol. 71, No. 3, pp. 409-419 (March 1983).
3. G. Herman, *Image Reconstruction from Projections*, Academic Press, New York.

on of a Source
ntific Labora-

um-Entropy
! (September

Algorithm
mputer

Handwritten notes:
① A neural network Computational
HAP Approach
Spense
② Artificial neural network
Adaptive System Applied
Bowman

ARTIFICIAL NEURAL NETWORK ADAPTIVE SYSTEMS APPLIED TO MULTISENSOR ID

Christopher Bowman

Ball Systems Engineering Division
9605 Scranton Road, Suite 500
San Diego, CA 92121

Artificial Neural Network (ANN) technology, ascribed as the sixth generation computer, is progressing rapidly in both devices and architecture designs. ANN's are being applied to a variety of problems with particular success in solving well-defined adaptive pattern recognition problems. The results are achieved faster and in some cases more accurately than existing numeric and symbolic approaches. This paper describes what ANN's are and how they are trained. A taxonomy is given along with ANN dynamics and training equations. Current ANN workstation capabilities are summarized. The application of ANN's to multi-sensor ID is addressed including a stereo image matching ANN and a hybrid distributed approach to automatic target recognition (ATR).

ARTIFICIAL NEURAL NETWORK ADAPTIVE SYSTEMS APPLIED TO MULTISENSOR ID

Christopher Bowman

Ball Systems Engineering Division
9605 Scranton Road, Suite 500
San Diego, CA 92121

1.0 INTRODUCTION

Artificial Neural Networks (ANN's) are computers that learn how to solve problems based upon sample data and built-in learning mechanisms. In other words, ANN's are trained to identify on their own the key features which enable them to distinguish different patterns. An ANN can learn on-line in real-time, or can be trained off-line by a user with a sample training set. The ANN's do not require expert knowledge representation, logical inferencing schemes, statistical algorithms, or a programmer to develop and code a solution to the user's problem. Also, ANN's do not provide step-by-step explanations as to how answers are achieved. ANN's do require an architecture with sufficient capacity and a training scheme. ANN's provide a complementary addition to conventional von Neumann processing for problems requiring pattern-recognition-type tasks.

This paper provides an introduction to what ANN's are and how they are trained in Sections 2 and 3. Section 4 summarizes their current capabilities. Potential applications to multi-sensor ID are presented in Section 5.

2.0 WHAT ARE ANN'S?

The human brain has roughly 10^{11} to 10^{13} neurons with 10 to 10^5 connections each. We learn by modifying these connections most of which are local. Neurons work in milliseconds and relax to a solution in less than a hundred steps (i.e., solutions in less than a second).

An ANN is a massively parallel inter-connected network of simple units intended to interact similarly to biological neural systems (Ref 14). As such ANN's are characterized as feedback dynamic systems which are (Ref 4):

- non-linear: node response functions
- non-local: nodes take outer products of their inputs

- non-stationary: time-independent node responses
- non-convex: numerous local minima

The ANN unit activation dynamics obey an autonomous dissipative system of non-linear differential equations,

$$\frac{dx_i}{dt} = F_i(X_1, \dots, X_n) \text{ for each unit } i \text{ activation level } X_i. \quad (1)$$

For any solution, $X(t)$, $X(t+c)$ is also a solution for all c . These paths in phase space do not intersect except where $F_i(X)=0$ for all i . Such points, X_c , are called critical or fixed points. A critical point is asymptotically stable when all paths in a neighborhood converge to it over time. Asymptotically stable critical points yield pattern recognition solutions when the appropriate F_i have been learned by the ANN. Three techniques that have been extensively used in ANN design to guarantee convergence to a fixed point are described next. First, Liapunov's theorem guarantees that a critical point is asymptotically stable if there exists a function E which is positive definite and whose derivative with respect to time is negative definite at that point. This is typically achieved via a symmetrically connected ANN. The second approach used to guarantee convergence is to require that the system be competitive (Ref 5),

$$\frac{dF_i}{dX_j} \leq 0 \quad \text{for all } i \neq j \quad (2)$$

(or cooperative by reversing the sign for all $i \neq j$ Ref 12). The third approach is to allow only feed-forward connections such as illustrated in Figure 2-1. The trained ANN classifies an external input, I , by settling initial state, X^0 , to the corresponding ANN system (i.e. $F(I, X)$) fixed point.

The objective in training the ANN is to modify the differential equations, specifically the F_i , to yield the correct classification (i.e., fixed points) for any allowed input. In first-order ANN's, the F_i are parameterized by Long Term Memory (LTM)

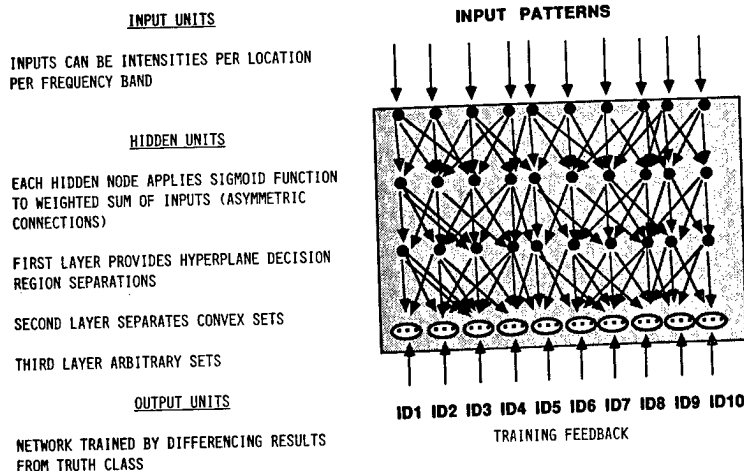


Figure 2-1. A Feed-Forward Neural Network

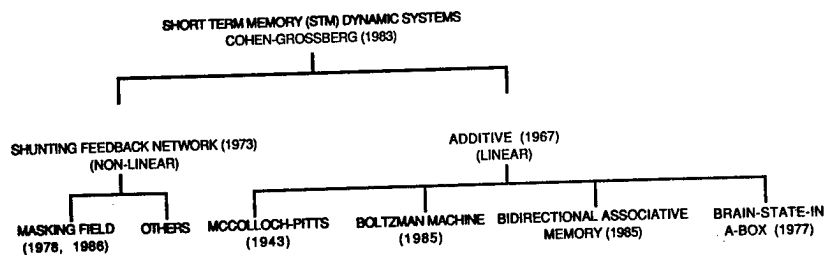
weights (W_{ij}) associated with the inputs to node i from other nodes j (including itself). In higher-order ANN's, these LTM weights (W_{ij}) depend upon combinations of node inputs: k . There are two approaches to learning these LTM weight parameters in ANN's: the weights may be trained in real-time (i.e., on-line), or they may be trained off-line. Once a fixed set of LTM weights have been determined, the ANN can be used as a Content Addressable Memory (CAM) network. A taxonomy of implemented CAM's (i.e., Short Term Memory (STM) dynamic systems) is shown in Table 2-1.

Sufficient fixed points need to be available to separate and store all the desired memories. Once this is accomplished, partial patterns (i.e. incomplete memories) are input usually via the initial ANN state, X^0 . The ANN then settles to the associated memory fixed point.

A useful activation function is the sigmoid

$$G_j(X_j) = \alpha_j + \left[1 + e^{-(X_j + \theta_j)/\tau_j} \right]^{-1} \quad (4)$$

Table 2-1. Content Addressable Memory (CAM) ANN Taxonomy (Ref 9)



The general STM dynamics model is given in Figure 2-2. Various ANN CAM systems may be defined by the type of units, connectivity, updating, and activations that are selected. For example, the Hopfield associative memory additive STM dynamics equation (Ref 13) is

$$C_i \frac{dx_i}{dt} = \sum_j W_{ij} V_j - X_i/R_i + I_i \quad (3)$$

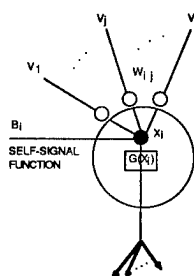
where W_{ij} is the weight from node j to node i
 V_j is the $G_j(X_j)$ output from node j
 R_i is the parallel sum of the resistances $1/W_{ij}$
 I_i are the external inputs to node i

where the α_j and θ_j effect the excitatory/inhibitory and threshold firing levels respectively. The τ_j determines the gain (i.e. slope) of the sigmoid which strongly effects the convergence properties of the CAM. Namely for large τ_j (i.e. low gain, G' small) the ANN will more likely converge. In fact, convergence is guaranteed (Ref 2) whenever

$$\left[\sum_i \sum_j W_{ij}^2 \right] (\max_i |G'_i|)^2 < 1 \quad (5)$$

For small τ_j (i.e. higher gain, G' large) the rate of convergence is faster. However, there may not be a fixed point and the number of bifurcations (i.e. alternating convergence points) tend to increase with decreasing τ_j causing "chaos". Although some conclusions may be drawn from the time weighted average of these alternating

INPUT SIGNALS FROM OTHER NODES



$$V_i = G_i(X_i) \text{ OUTPUT SIGNAL TO OTHER NODES}$$

Figure 2-2. Sample First Order Neural Network Node

SHORT TERM MEMORY (STM) DYNAMICS

$$\frac{dX_i}{dt} = A_i(X_i) [B_i(X_i) - \sum_{j=1}^N W_{ij} G_j(X_j)] = F_i(X_i)$$

(GROSSBERG CONTENT ADDRESSABLE MEMORY MODEL)

WHERE

LONG TERM MEMORY (LTM) DYNAMICS

$$\frac{dW_{ij}}{dt} = H_i(X_i) [D_{ij} G_j(X_j) - C_{ij} W_{ij}]$$

convergence points, CAMs with unique fixed points have been preferred.

3.0 HOW ARE ANN'S TRAINED?

A taxonomy of learning systems is given in Table 3-1. Two useful equations for real-time LTM weight dynamics as a function G_j of the unit inputs from the j th nodes and its outputs $H_i(X_i)$ are as follows (Ref 9):

Hebbian Learning (With Passive Decay)

$$\frac{dW_{ij}}{dt} = -C_{ij} W_{ij} + D_{ij} G_j(X_j) H_i(X_i) \quad (6)$$

Competitive Learning (Gated Decay Hebbian)

$$\frac{dW_{ij}}{dt} = H_i(X_i) [-C_{ij} W_{ij} + D_{ij} G_j(X_j)] \quad (7)$$

Under competitive learning the weight vector shifts towards the stimulus over time, whereas under Hebbian learning the weights increase with link firing caused by the stimulus. Another approach is differential Hebbian in which the weights increase with the rate of change in the link firing.

Off-line learning techniques are divided here into the following types:

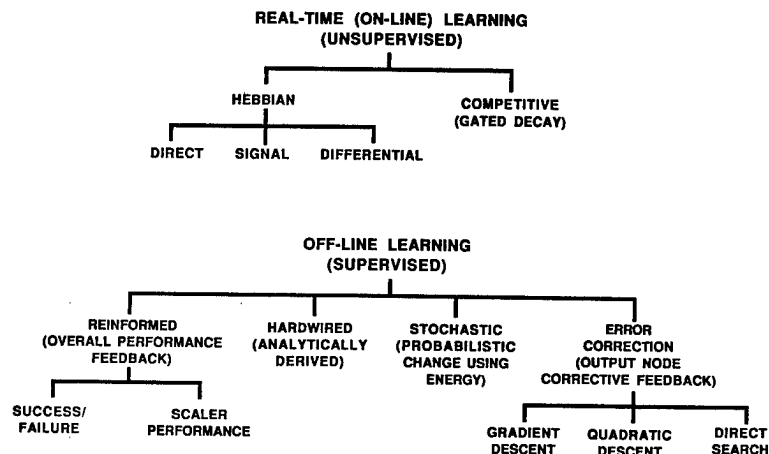
1. Error correction (corrective feedback per output node)
2. Reinforced (overall performance feedback)
3. Stochastic (probabilistic change using energy value)
4. Hardwired (analytically derived and fixed).

A popular error-correction technique is back-propagation (Ref 22,19,16) which is a gradient-descent technique usually applied to first-order, feed-forward networks. Back-propagation (i.e., delta-rule) training, as described in Figure 3-1, uses a steepest descent on the error-squared function over weight space. Namely,

$$\min(E^n) = 1/2 \sum_i \left[p_i^n - G\left(\sum_j W_{ij} v_j^n\right) \right]^2 = (\text{error})^2 \text{ for } n^{\text{th}} \text{ pattern } p^n \quad (8)$$

$$\frac{dE^n}{dW_{ij}} = - \left[p_i^n - G\left(\sum_j W_{ij} v_j^n\right) \right] \frac{dG_i(X_i)}{dX_i} v_j^n = \text{gradient change for } W_{ij} \quad (9)$$

Table 3-1. Long Term Memory (LTM) Learning System Taxonomy



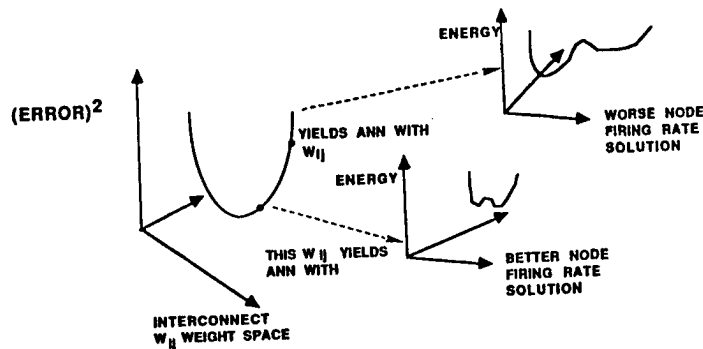


Figure 3-1. Back-Propagation (Delta-Rule) Supervised Learning)

where

$$X_i = \sum_j^N W_{ij} V_j \quad (10)$$

This has been shown by Lapedes and Farber (Ref 15) to be mathematically equivalent to minimizing a Liapunov energy function for a master N^2 -node symmetric ANN whose weights are already known from the original ANN being trained (i.e., the slave ANN), see Figure 3-2. Namely, the energy function for symmetric master ANN is

$$\min - \sum_{i,j,k,l}^N W_{ijkl} V_{ij} V_{kl} - \sum_{i,j}^N I_{ij} V_{ij} \quad (11)$$

where the symmetric weights are:

$$W_{ijkl} = -4 \sum_n^M \delta_{ki} V_{jn}^n V_{jl}^n \quad (12)$$

The V_{ij} fixed points yield the W_{ij} for slave ij network. The master network convergence rate is $O(N^3)$ which is typically slower than straight back-propagation. However with analog ANN's it may be preferred due to settling time requirements for back-propagation.

Pineda, Ref 20, has extended back-propagation to recurrent and higher-order ANN's. A recurrent network

allows feedback within each layer, and a higher-order ANN is of the form:

$$\frac{dX_i}{dt} = A_i(X_i) \left[B_i(X_i) - \sum_j \sum_k W_{ij}^{(n)} G_j(X_j) \dots G_k(X_k) \right] \quad (13)$$

where there are $n + 1$ indices and the summations are over all the indices except i . The n superscript is the order of the correlation of the network. Pineda's extension applies to case where $A_i(X_i) = -1$ and $B_i(X_i) = X_i + I_i$, and basically sets

$$\frac{dw_{ij}}{dt} = -C_w \frac{d}{dw_{ij}} (\sum (\text{Error})^2) \quad (14)$$

where C_w defines a time scale which is sufficiently slow that X is always essentially in steady state (i.e. weights change adiabatically). The solution to this linear equation for the weights is then recognized to be the fixed point of another dynamical system of transpose size which will converge if the original system converges (Ref 1). The resultant weight update requires $O(N^2)$ operations for N nodes.

Second-order schemes (e.g., Ref 18) try to avoid the traditional optimization problems of gradient descent by using the

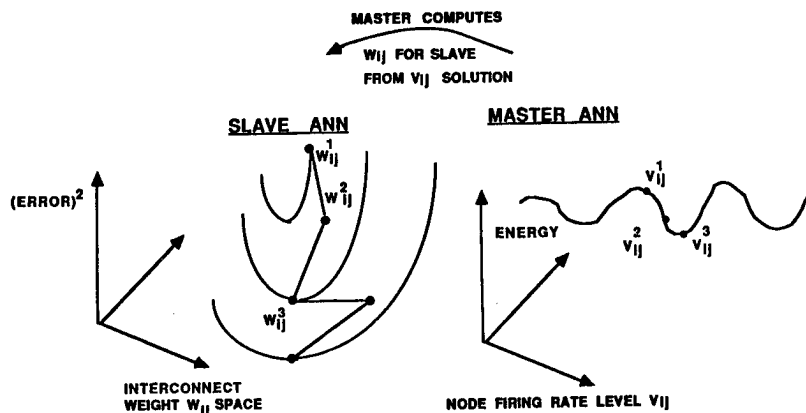
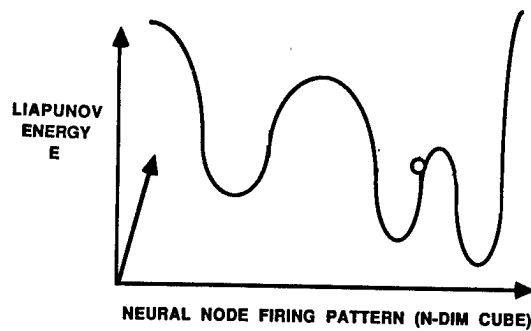


Figure 3-2. Master Network Computes Weights for Slave Network*



- LIAPUNOV FUNCTION DETERMINES WEIGHTS

$$E = -1/2 \sum_{I,J} \sum_{N} W_{IJ} V_I V_J - \sum_{J} I_J V_J + \sum_{I} \int_{R_{10}}^Y G_{I1}^1(V) dV \text{ FOR SYMMETRIC } W_{IJ} \text{ (HOPFIELD)}$$

- SIMULATED ANNEALING IS USED TO STIMULATE THE NETWORK TO ESCAPE LOCAL MINIMA

Figure 3-3. Ann with Hardwired Weights Relaxes to Solution

Hessian of the error function with respect to weight space. Direct search techniques determine a descent direction based upon numerous performance values (e.g., typically $N + 1$ points in N -space. They do not take analytical derivatives and are expected to perform better for very rough error functions by passing over many local minima. This approach can be followed by a gradient descent approach much as Hebbian learning has been followed by gradient descent in some applications.

Hardwired weights have been used when an optimization problem has been formulated in terms of an unconstrained minimization of a Liapunov function. The Liapunov function used with the Hopfield associative memory is shown in Figure 3-3 (Ref 13). This approach has been used in the Hopfield solution to the traveling salesman problem and in the P. Castelaz or H. Szu (Ref 24) solutions to the internetted report association problem illustrated in Figure 3-4.

Simulated annealing is typically applied to keep the ANN from settling into local minima. Such an approach has been proposed in preference to traditional approaches for very large problems (e.g., $N \gg 100$). For smaller problems, for example, a 30-city traveling salesman problem where the optimum route length is 4.3, the Hopfield ANN has converged to values of only seven or less whereas a random tour typically has a length of twelve.

Hardwired weights have also been used for non-symmetric ANN's to solve the linear programming problem. The problem constraint matrix becomes the weights between a column of the primal and a column of dual (i.e. price) variable nodes with the other variables as external inputs (Ref 6).

Reinforced learning is applied where error correction per output node is not available and only a success/failure

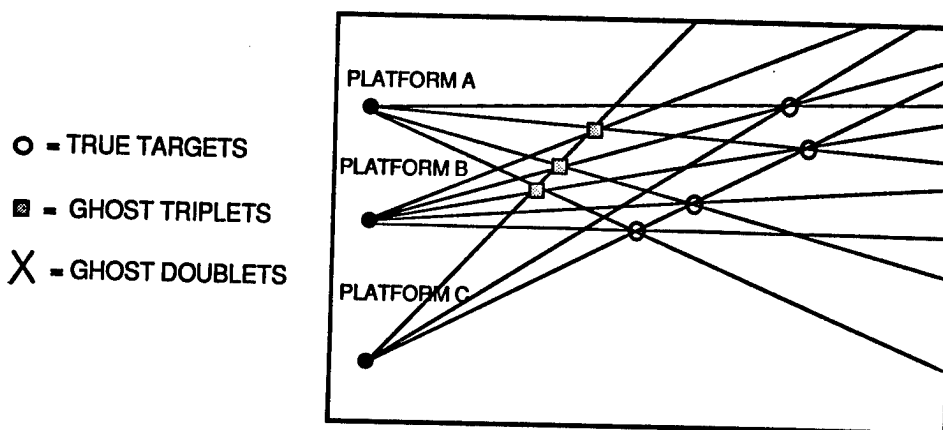


Figure 3-4. ANN Hardwired Weights Have Been Applied to Internetted Association Problem

response or scalar performance measure is available. A general equation for reinforcement learning is

$$W_{ij}(t+1) = W_{ij}(t) + \eta (r(t) - \theta_j) e_{ij}(t), \quad (15)$$

where η is a constant between 0 and 1 used to regulate the rate of learning, $r(t)$ is the success/failure scalar value provided by the environment at time t , θ_j is the reinforcement threshold value for the j th output unit, and $e_{ij}(t)$ is the canonical eligibility of the weight between the i th and the j th unit at time based upon the probability that the computed output value equals the desired output value (Ref 25,26).

4.0 WHAT ARE CURRENT ANN CAPABILITIES?

A description of the top commercial ANN hardware available to date is given in Table 4-1. Thus current ANN hardware supports thousands of nodes which theoretically can yield memory capabilities also in the thousands. This can be derived for the ANN STM given by Eq. 3 as follows. (Ref 11). Given M patterns as fixed points yields NM linear equations in $N^2 + N$ unknowns W_{ij} and R_i . Requiring asymptotic stability at these M fixed points yields an additional $2NM$ piecewise linear inequalities from Gershgorin's theorem. Thus independent ANN parameters exist yielding $M = N+1$ stable equilibria patterns for N nodes. However, the observed capacity for many ANN systems has been less (e.g. approximately $.14N$, (Ref 13,15,22)).

ANN capacity performance is continually improving. In fact, ANN's have demonstrated pattern recognition performance superior to existing numeric and symbolic approaches for sonar returns (Ref 8) and protein strings (Ref 21). The payoff for ANN's depends upon innovative ANN STM dynamics and LTM training designs implementable in cost effective "analog" hardware.

5.0 PROMISING ANN MULTI-SENSOR ID APPROACHES

Multi-sensor ID is the key missing technology necessary to support the Beyond Visual Range (BVR) attack required for US fighters to win against a superior number of threats. The multi-sensor ID function, a subfunction of the multi-sensor integration (MSI) function, uses the track kinematics and attribute data association decisions to identify which attributes should be considered with each track from each sensor, as shown in Figure 5-1. Multi-sensor ID uses these inputs plus a priori data to generate the MSI target ID results to support the pilot interface and asset management.

A key problem associated with multi-sensor ID has been that the individual sensors have difficulty generating their ID declarations with accurate error descriptors/statistics so that MSI can appropriately combine conflicting and uncertain ID data. The multi-sensor ID problem requires an innovative approach to overcome the following obstacles:

- long solution time expected to handle sundry multi-attribute ID and a priori data.
- inherent instability and errors in the attribute data which are not well-characterized (as a result, algorithms for clustering ID discriminants are ill-defined, necessarily complex, and hard to develop); and
- requirement for fault tolerance in the implemented solution technique and hardware architecture.

ANN approaches appear promising due to their demonstrated potential for the following:

- fast settling (milliseconds) to "good" solutions since the long term memory is imbedded in the unit connection weights with no von Neumann bottleneck between memory and processor;

Table 4-1. Top Commercial Ann Hardware Available (1988)

	MAXIMUM NUMBER OF TOTAL INTERCONNECTS AND NODES (MILLIONS)	HOST	INTER-CONNECTS PER SEC W/NO TRAINING (MILLIONS)	WORKSTATION COST
TRW MARK V (8 68020 BOARDS)	5	VAX	5/10	\$70K
SAIC Σ -1 (DELTA PROCESSOR)	3.1	PC/AT (80386/7)	2.7/10	\$25K
HNC ANZA PLUS (WEITEK ACCEL)	2.5	PC/AT (ZENITH 386-80)	1.5/6	\$25K

*AN N-NODE ANN USES UP TO N^2 INTERCONNECTS

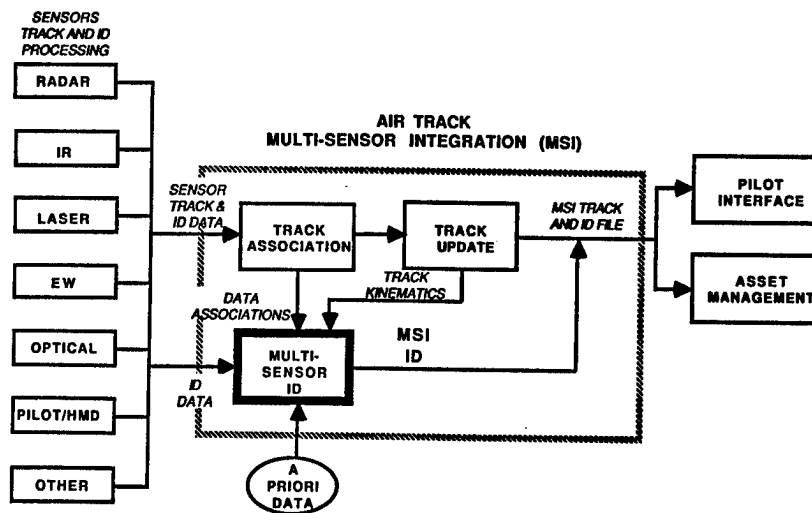


Figure 5-1. Multi-Sensor ID is the Toughest Subfunction of MSI

- self-organized learning which automatically recognizes ID discriminant patterns based upon training data
- highly fault-tolerant since it is hardware (not software) intensive and redundant.

These inherent ANN properties address the technical obstacles to successful development of a multi-sensor ID avionics capability identified above. The ANN technology is ripe now for multisensor ID performance demonstrations using one of the many flexible ANN workstations commercially available.

Multi-sensor ID fusion can take place at many levels as depicted in Figure 5-2. Higher level fusion is desired to minimize cost/complexity by allowing individual sensor processing. Lower level fusion can improve performance by increasing target detection and classification confidence and range. As always the level of fusion design depends upon the problem, requirements, and available technology.

One problem for which pixel level fusion has been suggested (Ref 7) is the fusion of TV and FLIR images. The TV is strong on determining the boundaries of objects whereas the FLIR photometric data is useful for extracting features. ANN's are potentially useful in supporting the corresponding "boundary contour system", "feature contour system", and the "object recognition system" (Ref 10). Another role for ANN's in pixel-level fusion is in solving the image co-registration (or stereo image matching) problem. The crux of this problem is to determine the matching pixels in two images or equivalently to decide what the disparity is between one image and another for each pixel. A potential ANN to solve this problem, as depicted in Figure 5-3 (Ref 17), uses many layers each representing a different disparity level. The resulting max a posteriori energy function for this recursive ANN contains image compatibility, smoothness via Markov Random Fields (MRF), and uniqueness constraints on the solution. Improvements on this basic approach are given in Ref 3.

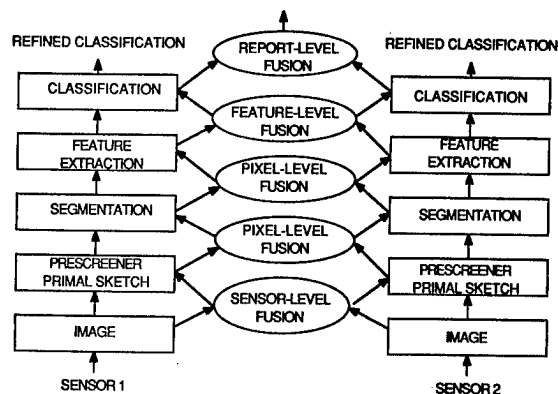


Figure 5-2. Levels of Multi-Sensor ID Fusion

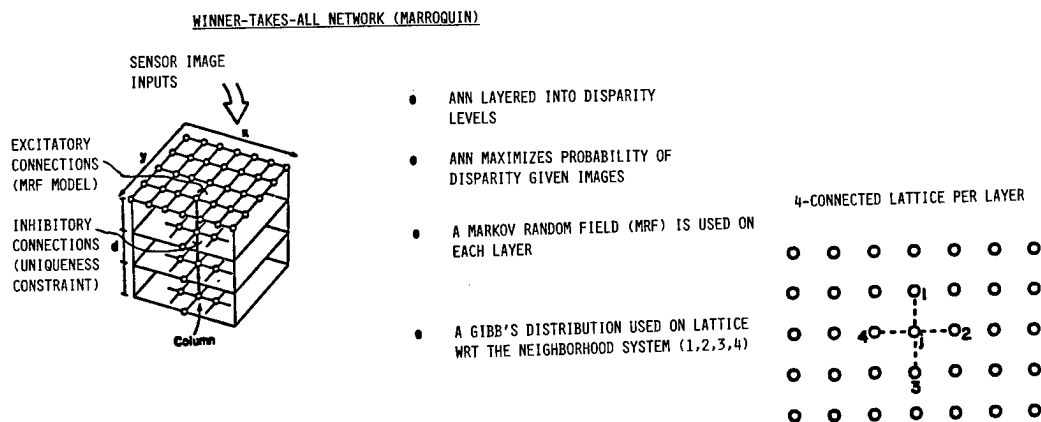


Figure 5-3. 3-D ANN Image Co-Registration

ANN's can raise the knee-of-the-curve design level towards more decentralization by improving the target classification speed and accuracy of individual sensors. This increased parallelization will reduce the complexity and time delays associated with more centralized approaches. ANN technology may be useful as a preprocessor to set-up the problem parameters for traditional Automatic Target Recognition (ATR) software. Such hybrid systems are envisioned due to the complementarity of ANN and conventional processing as described in

Figure 5-4. An example of such a decentralized hybrid application of ANN's with numeric and symbolic (e.g. expert system) processing is shown in Figure 5-5. Here four layer feed-forward ANN's (with recursion within layers) are applied separately to radar, laser, IR, or EO image or spectra data as appropriate and trained off-line to support target ID. Traditional schemes (e.g. log-polar or Fourier transforms) may be used to accomplish rotation, scale, or shift-invariance transformations. However, these may be augmented by ANN's to pull in 3-D information.

	KNOWLEDGE REPRESENTATION	GROWTH IN PROBLEM COMPLEXITY	PROBLEM SOLVING	SOLUTION CONVERGENCE	RELIABILITY	EXECUTIVE CONTROL	LONG-TERM KNOWLEDGE	SYSTEM DEVELOPMENT COSTS	EFFECTIVE PROBLEM DOMAINS
MODESTLY PARALLEL NUMERICAL AND SYMBOLICS PROCESSING	REPLICATED LOCAL REPRESENTATIONS	MORE SOLUTION TIME AND MEMORY	SOLUTION TECHNIQUES PROGRAMMED SO MORE EFFICIENT	TIMELINESS DEGRADES WITH ACCURACY	FAULT RECOVERY PLANNED	RULE-DRIVER TOP-DOWN	IN LARGE DATABASES	MORE SOFTWARE INTENSIVE	KNOWN FEATURES AND STATISTICS
MASSIVELY PARALLEL DISTRIBUTED "NEURAL" PROCESSING	DIFFERENT DISTRIBUTED REPRESENTATIONS	NEW LARGER NETWORKS REQUIRED	LEARNS SOLUTION VIA SOLUTION EXEMPLARS	TIMELY APPROXIMATE SOLUTIONS	INHERENTLY FAULT TOLERANT	SELF-ORGANIZED BOTTOM-UP	DISTRIBUTED IN CONNECTION WEIGHTS	MORE HARDWARE INTENSIVE	UNKNOWN FEATURES AND STATISTICS

Figure 5-4. Complementarity of ANN and Conventional Processing

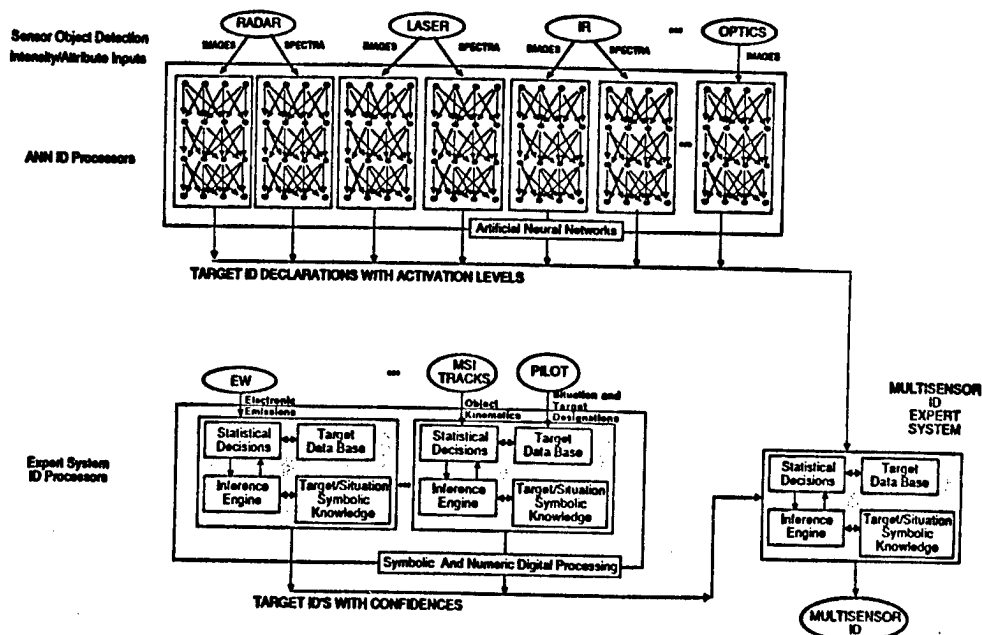


Figure 5-5. Distributed Hybrid Application of Artificial Neural Networks with Expert Systems for Cost Effective NCTR

REFERENCES

1. L. B. Almeida, A learning rule for asynchronous perceptrons with feedback in a combinatorial environment, in "Proceedings of the IEEE First Annual International Conference on Neural Networks", (M. Claudil and C. Butler, eds.), pp. 609-619, San Diego, California (1987).
2. A. A. Atiya, Learning on a general network, to appear in "Proceedings of IEEE Conference on Neural Information Processing Systems", (Dana Z. Anderson, ed.), Denver Colorado, Nov. 8-12 (1987).
3. C. L. Bowman, "Artificial Neural Network Conceptual Designs for Image Coregistration," BSED R-2-88, Feb. 1988.
4. Y. T. Chien, J. Deken, A. Harvey, IEEE First International Conference Neural Networks Proceedings June 1987.
5. M. A. Cohen, S. Grossberg, "Absolute Stability of Global Pattern Formation and Parallel Memory Storage by Competitive Neural Networks," IEEE Transactions on Systems, Man, and Cybernetics, SMC-13, 815 - 825 (1983).
6. J. C. Culsolli, V. Protopopescu, "A Linear-Programming Method Inspired by The Neural Networks Framework, CESAR-88/22, July 1988.
7. G. Duane, "Pixel-level Sensor Fusion for Improved Object Recognition," SPIE Conference April 1988.
8. R. P. Gorman, T. J. Sejnowski, "Analysis of Hidden Units in a Layered Network Trained to Classify Sonar Targets," Neural Network Journal Volume 1, November 1987.
9. S. Grossberg, "Nonlinear Neural Networks: Principles, Mechanisms, and Architectures," Neural Networks, Vol. 1, pp. 17-61, 1987.
10. S. Grossberg, "Cortical Dynamics of Three-Dimensional Form, Color, and Brightness Perception: I. Monocular Theory," Perception and Psychophysics, 41(2), 87-116 (1987).
11. A. Guez, V. Protopopescu, J. Barhen, "On the Stability, Storage Capacity and Design of Nonlinear Continuous Neural Networks," Oak Ridge National Laboratory (ORNL) ORNL/TM-10329, CESAR-86/49, and CSNP-86/01.
12. M. W. Hirsch, Systems of differential equations that are competitive or cooperative, II: Convergence almost everywhere. SIAM Journal of Mathematical Analysis, 16, 423-439 (1985).
13. J. J. Hopfield, "Neurons with graded response have collective computational properties like those of two-state neurons," Proc. Nat. Acad. Sci. USA, Bio. 81, 3088 - 3092, (1984).
14. T. Kohonen, "An Introduction to Neural Computing," Neural Networks Journal Vol 1. (November 1987).

15. A. Lapedes, R. Farber, "A self-optimizing, nonsymmetrical neural net for content addressable memory and pattern recognition, *Physica*," D22, 247 - 259, 1986, see also "Programming a Massively Parallel, Computation Universal System: Static Behavior, in *Neural Networks for Computing Snowbird*," UT 1986, AIP Conference Processing, 151, edited by John S. Denker (1986)
16. Y. LeChun, A learning procedure for an asymmetric threshold network, *Proceedings of Cognitiva*, 85, p. 599, (1985).
17. J. Marroquin, S. Mitter, T. Poggio, Probabilistic Solution of Ill-Posed Problems in Computational Vision. *American Statistical Assoc., Journal of the American Statistical Assoc.*, Vol. 82, No. 397, Theory and Methods (1987).
18. D. B. Parker, "Second Order Backpropagation: Implementing an Optimal $O(n)$ Approximation to Newton's Method as an Artificial Neural Network," submitted to *Computer*, (1987).
19. D. B. Parker, "Learning-Logic, Invention Report," S81-64, File 1, Office of Technology Licensing, Stanford University, October, (1982).
20. F. J. Pineda, "Generalization of Backpropagation Recurrent and Higher Order Neural Networks." To appear in the *Proceedings of IEEE Conference on Neural Information Processing Systems*, Denver, Colorado, Nov. 8-12, (1987).
21. N. Qian, T. J. Sejnowski, "Predicting the Secondary Structure of Globular Proteins Using Neural Network Models," September 21 1987.
22. D. E. Rumelhart, G. E. Hinton, R. J. Williams, in *Parallel Distributed Processing*, edited by D. E. Rumelhart and J. L. McClelland, M.I.T. press, (1986).
23. P. K. Simpson, "A Review of Artificial Neural Systems I: Foundations", Submitted to CRC in AI (1988).
24. H. Szu, Non-convex optimization, *Proceedings of the SPIE: Real time signal processing IX*, 698, 59, (1986).
25. B. Widrow, N. Gupta, S. Maitra, Punish/reward: Learning with a critic in adaptive threshold systems, *IEEE Transactions on Systems, Man, and Cybernetics*, SMC-5, 455, (1973).
26. R. Williams, Reinforcement-learning connectionist systems, Northeastern University, College of Computer Science Technical Report, NU-CCS 87-3, (1987).

THIS PAGE LEFT BLANK INTENTIONALLY

A NEURAL NETWORK COMPUTATIONAL MAP APPROACH TO SENSORY FUSION

Clay D. Spence¹, John C. Pearson¹, Jack J. Gelfand¹, W.E. Sullivan² and Richard M. Peterson¹

1. David Sarnoff Research Center, Subsidiary of SRI International,
Princeton, New Jersey 08543-5300

2. Department of Biology, Princeton University,
Princeton, New Jersey 08544

ABSTRACT

We present a neural network model for sensory fusion based on the design of the visual/acoustic target localization system of the barn owl. This system adaptively fuses its separate visual and acoustic representations of object position into a single joint representation used for head orientation. The building block in this system, as in much of the brain, is the neuronal map. The acoustic processing system of the barn owl uses them to form a map of space on the structure known as the external nucleus of the inferior colliculus. This acoustic map, and the visual map of the retina, jointly project onto the optic tectum, creating a fused visual/acoustic representation of position in space that is used for object localization. In this paper we describe our mathematical model of the process which maintains the registration of the visual and acoustic inputs to the optic tectum. The model assumes that the acoustic projection from the external nucleus onto the tectum is roughly topographic and one-to-many, while the visual projection from the retina onto the tectum is topographic and one-to-one. A simple process of self-organization alters the strengths of the acoustic connections, maintaining a small region of strong acoustic connections whose inputs are coincident with the visual inputs. Computer simulations demonstrate how this mechanism can account for the existing experimental data on adaptive fusion and makes sharp predictions for experimental test.

A NEURAL NETWORK COMPUTATIONAL MAP APPROACH TO SENSORY FUSION

Clay D. Spence¹, John C. Pearson¹, Jack J. Gelfand¹, W.E. Sullivan² and Richard M. Peterson¹

1. David Sarnoff Research Center, Subsidiary of SRI International,
Princeton, New Jersey 08543-5300

2. Department of Biology, Princeton University,
Princeton, New Jersey 08544

INTRODUCTION

Neural network research is largely oriented towards solving "high-level" problems such as pattern recognition, categorization, and associative memory. Recent developments in this field have produced powerful new architectures and algorithms for solving such problems (see review by Cowan and Sharp (Ref. 1)). Due to our relative ignorance of how such problems are solved by animals, these "high-level" neural networks are only loosely based upon known principles of brain organization and function, and do not directly correspond to any known brain structures. However, there are other equally important problems for which the corresponding brain structures are well characterized. Object localization and identification is one such problem, and sensor fusion plays an important role in the brain's solution to this problem. In this paper, a neural network approach to sensor fusion is presented that is based upon the map-like brain structures that solve the acoustic object localization problem in the Barn Owl. Neuronal maps are key building blocks of nervous system function, ranging from perceptual classification to motor control. These structures consist of locally interconnected arrays of neurons whose response properties vary systematically with position in the array, thus forming a map-like representation of information. Computation is achieved through transforming the representation from one map to the next. The fidelity of these transformations is maintained through dynamic processes of self-organization, endowing them with self-optimizing and fault tolerant properties. These structures are linked together in modular, hierarchical processing systems, that employ some of the same problem solv-

ing approaches used in technical applications, such as sensor fusion. In this paper we first briefly describe the stages in the chain of neuronal processing that generate a map of space from acoustic timing cues, and adaptively fuses it with the map of space derived from the retina. We then describe our proposed neuronal mechanism for the stage of visual/acoustic fusion and present results of computer simulations. This mechanism exploits the coincident signals produced by an object in the visual and acoustic representations, using adaptive, non-linear, neuron-like processing elements.

BARN OWL VISUAL/ACOUSTIC FUSION SYSTEM

Behavioral and physiological studies have revealed that owls use interaural intensity cues to specify the elevation of sounds, and interaural timing cues to localize the azimuthal direction (Refs. 2,3). The neuronal processing leading to azimuthal sound localization and visual fusion is accomplished by a series of four so called "computational maps", as shown in Fig. 1, and reviewed by Knudsen (Ref. 4). The processing for elevation follows a similar design and is omitted from the present discussion for clarity and brevity.

Nucleus laminaris (N. lam.) generates a map of interaural phase delay vs. frequency given phase-locked input signals from the cochlear nucleus. The central nucleus of the inferior colliculus (ICc) transforms the N. lam. map into a map of frequency vs interaural delay. The external nucleus of the inferior colliculus (ICx) transforms the ICc map into a map of space, forming an "acoustic retina." The acoustic space map (ICx) and the visual space map (retina) are fused in their

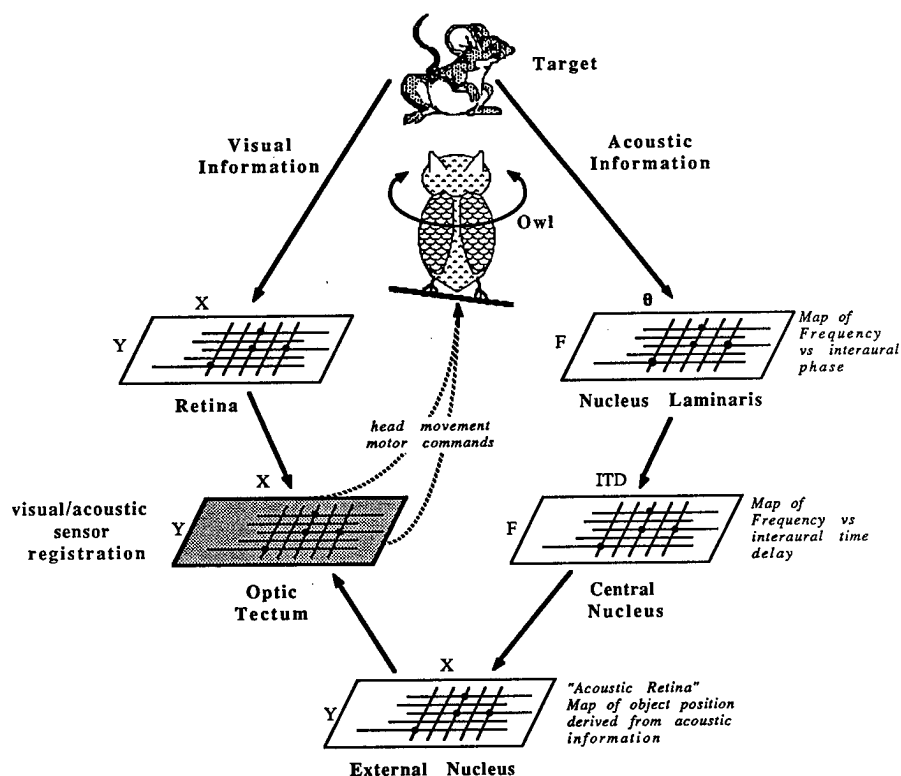


Figure 1. Illustration of the series of transformations in neuronal representation that produce adaptive visual/acoustic fusion in the object localization system of the barn owl. Only the azimuthal acoustic system is shown for clarity. A similar parallel series of transformations between neuronal arrays computes elevation.

joint projection onto the optic tectum. This fused map of object location is then used in orienting the head to center the object in the visual field for closer scrutiny.

The neuronal processes of visual/acoustic fusion are known to be adaptive during the growth period of the owl (Ref. 5). This is essential for the young owl, for during this time the distance between the ears increases severalfold, and this distance is a critical parameter in computing azimuthal position. Laboratory experiments have shown that perturbations to either the visual or auditory transducers (e.g., goggles or ear plugs) initially cause registration errors between the auditory and visual space maps. As a consequence, head orientation driven by acoustic cues fails to center the object in the visual center of view. However, with time, fusion is reestablished, and proper localization behavior is restored in a continuous manner.

The fused sensory map of space in the optic tectum is also a motor map that orients the head to center objects in the visual field, as illustrated in Fig. 2.

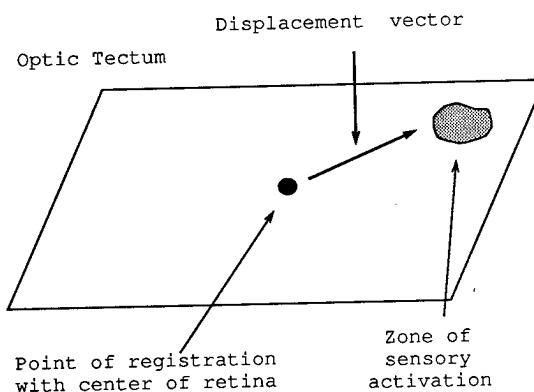


Figure 2. Representation of motor command for head orientation by position of activation on neuronal map in head centered retinotopic coordinates.

The static topographic projection of the retina onto the tectum, and the fact that the eyes do not move relative to the head, establishes head centered retinotopic coordinates on the tectum. Because the acoustic map (ICx) is fused with the visual map, the same region of the tectum will be activated by either visual or acoustic signals from a particular location in space. A vector from this point to the point representing the center of the retina represents the magnitude and direction of the head movement necessary to bring this source to the center of the visual field. Possible neuronal mechanisms for this have been presented by Grossberg and Kuperstein (Ref. 6) for the similar case of saccadic eye movements. Multiple objects presumably create multiple regions of activation on the tectum, and the system must select a single target for the head orientation response. Didday (Ref. 7) and Arbib (Ref. 8) have presented neuronal models of this function for the case of the frog.

NEURONAL MECHANISM FOR SENSORY FUSION

Recent work by Pearson, Finkel and Edelman (Ref. 9) demonstrated a solution to a problem related to the neuronal map fusion problem. They modeled the cortical map of touch sensation of the hand, which contains a topographic representation of both front and back hand surfaces. The representation of these two surfaces is not fused. Each cell in the map responds to stimulation of only one surface, and cells with the same preference are clustered into regions that are separated from regions with the opposite preference by sharp borders. Experiments have shown (see references within Ref. 9) that the borders between these regions dynamically shift so that more highly stimulated regions of the hand have larger regions of representation in the map and greater resolution. In the model, each cell received equal numbers of connections from corresponding regions of both surfaces. A rule for changing the strengths of the connections strengthens inputs that are spatially and temporally correlated, and weakens those that aren't. Since the two surfaces are rarely stimulated at the same time, cells weaken their connections with one surface while strengthening their connections to a small, compact region of the other.

Our model for fusion in the tectum is a simplified version of this model of the map of the hand. The front and back surfaces of the hand correspond to the visual and acoustic space maps. Fusion is produced, instead of segregation, because stimulation of corresponding regions of the two input maps is correlated rather than uncorrelated. Fig. 3 is a pictorial representation of the flow of signals from

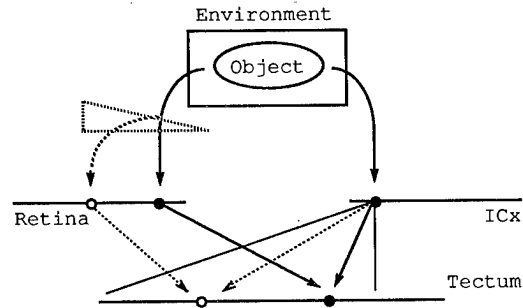


Figure 3. Proposed neuronal architecture for adaptive registration of visual (retina) and acoustic (ICx) neuronal maps of object position in the tectum of the Barn Owl following perturbation to the visual input (prism).

an object in the environment through the visual and acoustic space maps to a cell (marked with a filled dot) in the optic tectum. The light lines from the marked ICx cell to the tectum delimit its divergent region of projection, while the heavy line from the ICx cell to the tectum represents the functional projection created by strengthening that subregion of the total projection and weakening all others. This divergent projection is an assumption of the model. The dashed prism indicates schematically what would happen if there is a distortion added to the visual field. Immediately after the perturbation, the visual and acoustic maps in the tectum are out of register. The new point of activation in the retina (marked with an unfilled dot) immediately leads to the activation of a different cell in the tectum (marked with an unfilled dot) whereas the acoustic input fires the same cell in the tectum as before (filled dot). As a result, a single object will activate two cells in the tectum instead of one, and the input to a tectal cell will be half of what it was before the perturbation. However, with sufficient correlated stimulation of the visual and acoustic input maps, the connection strengths are altered so as to strengthen those acoustic connections that are coincidentally activated with the visual input, and to weaken the original acoustic connections that are no longer activated at the same time as the visual connections. To test these ideas a simplified computer model was constructed.

COMPUTER MODEL OF SENSORY FUSION

Fig. 4 shows preliminary results of a simulation of adaptive fusion following the type of perturbation to the visual input described in Fig. 3. These drawings are plots of the input connection strengths from the acoustic space map (ICx) onto the cell at the center of the fused map (tectum). The series of six drawings shows these connections as ini-

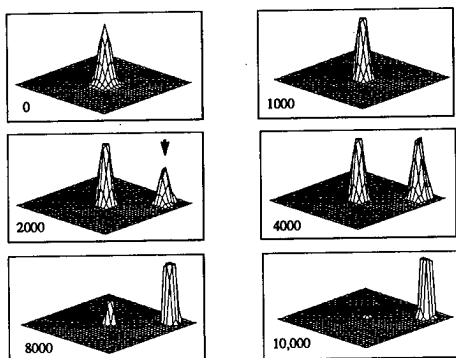


Figure 4. Simulation of reestablishment of visual/acoustic fusion following visual perturbation. Arrow indicates site of acoustic input coincident with visual input after the perturbation.

tially assigned (0), after 1000 time steps of unperturbed input, (1000), and at four successive times after the perturbation that moved the center of coincident input to the upper right as indicated by the arrow.

In this simulation a visual/acoustic stimulus 3x3 grid points in size was applied in a random sequence over the entire input grid, coincidentally activating topographically corresponding points in the space maps of the retina (R) and ICx (X). The tectal cell received a visual input from one fixed location (R0) that was activated when the stimulus covered it. The stimulus generated input was summed to yield the cell's potential, v , shown in Equ. 1.

$$v = c_j X_j + c_0 R_0 \quad (1)$$

where:

c_j = variable acoustic connection strength, ($0 < c_j < 1$)

c_0 = fixed visual connection strength, ($c_0=10$)

X_j = acoustic input, ($X_j \{0,1\}$, 1=on, 0=off)

R_0 = visual input, ($R_0 \{0,1\}$, 1=on, 0=off)

The acoustic connection strengths were then modified according to a simplified version of the synaptic rule proposed in Ref. 9, shown in Equ. 2.

$$c_j(t+1) - c_j(t) = k s(v, q_1, e_1) X_j - d s(v, q_2, e_2) |1 - X_j| \quad (2)$$

where:

k = growth constant, ($k=.05$)

d = decay constant, ($d=.05$)

s = sigmoidal function,

$$s(v, q, e) = [\exp((q-v)/e) + 1]^{-1}$$

q = threshold parameter, ($q_1=9.5$, $q_2=16$)

e = sharpness parameter, ($e_1=.01$, $e_2=.01$)

Parameters q_1 and q_2 set the thresholds for significant strengthening and weakening, respectively. Significant strengthening of an acoustic connection requires that its input be active ($X_j=1$) and that enough other strong connections (whether visual or acoustic) be active so that $v > q_1$. Weakening an acoustic connection requires that its input be inactive ($X_j=0$) and that enough other strong connections be active so that $v > q_2$. Given the fixed value of c_0 (arbitrarily chosen) and the stimulus size (chosen based on simple considerations of scale), q_1 was set so that weak acoustic connection strengths would grow very slowly unless they were activated coincidentally with the visual input, and q_2 was set so that only the coincident activation of both strong visual and acoustic inputs would produce weakening in the inactive acoustic inputs. Parameters k and d simply determine the time scale or "smoothness" of the simulation.

The reestablishment of fusion is robust to changes in the parameters, as long as the above guidelines are met. Larger stimuli simply enlarge the region of the strong acoustic connections. Larger values of e soften the non-linearity in s , making it easier to strengthen and weaken connections, but do not significantly affect the results. Regular stimulation, in which the stimulus moves over the input grid one point at a time, works as well as random stimulation. Changes in the threshold parameters affect the rate at which the original peak decays and the new peak grows, but not the final outcome, because the region of the new peak has the advantage of a fixed visual input. Of course, q_1 must be less than the voltage due to the visual input, R_0 , or the new peak cannot grow, and q_2 must be less than the maximal potential.

The model makes several biological predictions that could be tested. The model assumes a topographic, divergent projection from the ICx to the tectum (see Fig. 3). This could be tested with various anatomical tracing methods. The width of the divergence sets the maximum range over which registration errors can be corrected. During the adaptation to the perturbation, the auditory responsiveness of cells in the tectum should change in a characteristic way. A new region of auditory responsiveness should appear along with the original region, and as the new region gains in strength the original region should weaken, eventually vanishing. At first, this appears to be in contrast with the behavioral result, which is that the localization error vector slowly decreases in magnitude. However, it is consistent with recent findings that the motor output is determined by the vector average of activity on the tectum (Ref. 10).

CONCLUSIONS

Study of the visual/acoustic localization system in the Barn Owl has disclosed a potential neuronal mechanism for adaptive multi-sensor registration. Computer simulations of a neural network model of this system have successfully tested the proposed mechanism and produced predictions for experimental testing. Future work must determine the suitability of this method for technical applications.

ACKNOWLEDGMENTS

This work was supported by internal funds of the David Sarnoff Research Center.

REFERENCES

1. J.D. Cowan and D.H. Sharp, "Neural nets", Daedalus, Winter (1988).
2. E.I. Knudsen and M.J. Konishi, "Mechanisms of sound localization by the barn owl (*Tyto alba*)", J. Comparative Physiology 133:13-21 (1979).
3. A. Moiseff and M.J. Konishi, "Neuronal and behavioral sensitivity to interaural time differences in the owl", J. Neuroscience 1:40-48 (1981).
4. E.I. Knudsen, S. du Lac and S.D. Esterly, "Computational maps in the brain", Ann. Rev. Neuroscience 10:41-65 (1987).
5. E.I. Knudsen, "Early auditory experience aligns the auditory map of space in the optic tectum of the Barn Owl", Science 222: 939-942 (1983).
6. S. Grossberg and M. Kuperstein, "Neural Dynamics of Adaptive Sensory-Motor Control," North-Holland, Amsterdam (1986).
7. R.L. Didday, "A model of visuomotor mechanisms in the frog optic tectum", Math. Biosciences 30:169-180 (1976).
8. M.A. Arbib, Visuomotor coordination: from neural nets to schema, in "Adaptive Control of Ill-Defined Systems", Sulfridge et.al., eds., Plenum Press, New York, 207-225 (1984).
9. J.C. Pearson, L.H. Finkel and G.M. Edelman, "Plasticity in the organization of adult cerebral cortical maps: a computer simulation based on neuronal group selection", J. Neuroscience 7(12):4209-4223 (1987).
10. S. du Lac and E.I. Knudsen, "The optic tectum encodes saccade magnitude in a push-pull fashion in the barn owl", Society for Neuroscience Abstracts, p. 393 (1987).

APPLICATIONS OF A CONDITIONAL EVENT ALGEBRA TO DATA FUSION

Dr. I.R. Goodman

Command & Control Department
Code 421
Naval Ocean Systems Center
San Diego, California 92152

Abstract

This paper, in the spirit of last year's article, is a contribution toward a unified theory of data fusion as an integral part of a more general C^3 theory. A computationally feasible and mathematically sound procedure is proposed here for the combination of disparate information, prior to, and compatible with, ordinary conditional probability evaluations. This is based directly upon a new breakthrough concerning the extension of classical probability logic to a full conditional logic.

1. DATA FUSION AND C^3 PROCESSES

1.1. Qualitative aspects.

Previously, in [1] a general approach to data fusion was outlined within the context of C^3 processes.

As a brief review (see also [2],[3]) and a modification of past efforts, the following obtains:

The author has been considering C^3 processes from the generic viewpoint of interacting nodes of decision-makers, or complexes of such. These interactions or "signals" may be actual vectors of signals containing voluntary or leaked information from other nodes, friendly or hostile, or they may represent fired weapons, for example. The nodes are relative in size, but whether they represent one or a group of individuals, they possess certain common characteristics. These include a decision structure centering around data fusion which contains detection, hypotheses forming mechanisms, algorithm selections, and responses as an output. The nodes also are represented by corresponding state vectors containing all pertinent descriptors, such as equations of motion and location, number of individuals or supplies and their relative importance, damage levels, threat levels, and estimates and other knowledge of other node states, friendly or adversary.

Schematically, the following simplified situations hold as Figures 1 and 2 show:

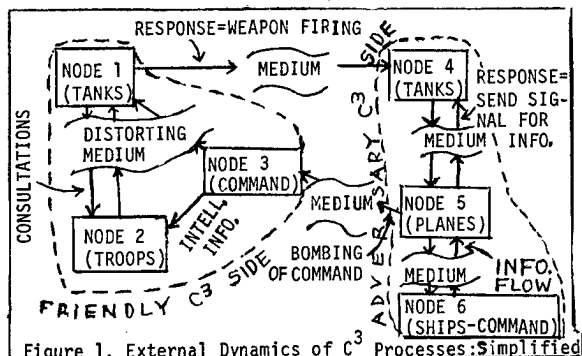


Figure 1. External Dynamics of C^3 Processes: Simplified

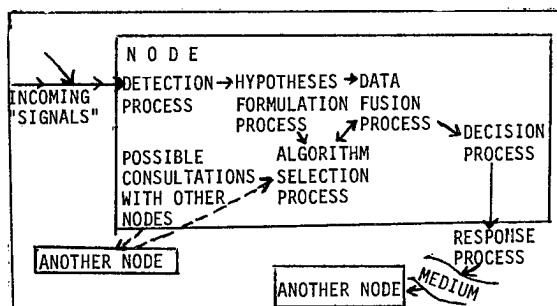


Figure 2. Internal Dynamics of C^3 Processes: Simplified

The basic evolution cycle of a typical node is described in the following figure:

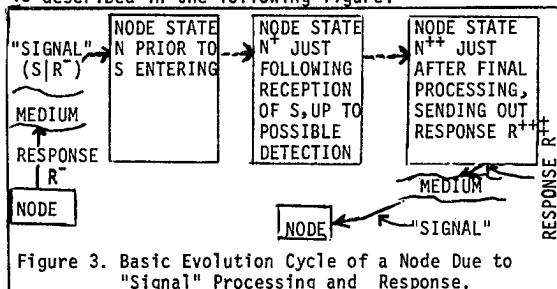


Figure 3. Basic Evolution Cycle of a Node Due to "Signal" Processing and Response.

The components of a typical node state are as in Table 1:

(NODE STATE VECTOR)	=	(NODE STATE PROPER KNOWLEDGE PART)	=	(THREAT LEVEL # OF TROOPS # OF WP, I # OF WP, II IMPORTANCE SUPPLY LEVEL EQ. OF MO. DAMAGE LEVEL ESTIMATES OF OTHER NODE STATES)
---------------------	---	------------------------------------	---	--

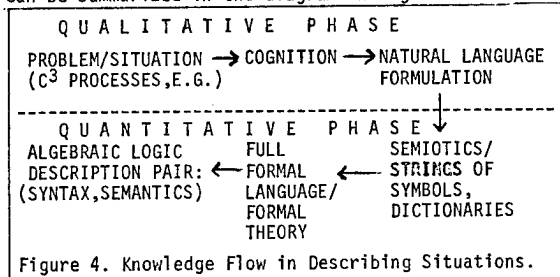
Table 1. Components of C^3 Node States.

1.2. Quantitative aspects.

The next step following the qualitative scoping-out of C^3 processes, including data fusion, is the determination of the corresponding quantitative description. In effect, this entails choosing both an appropriate relational syntax and a numerical/semantic evaluation system. Such a pair is called an *algebraic logic description pair* (ALDP). The reader is, most likely,

familiar with the most common ALDP: probability logic (PL) where the appropriate syntax is the structure of a boolean (or more strongly, sigma) algebra R of events. Here, the usual set or logical/propositional operations hold: unions(\cup) or disjunctions(\vee), intersections(\cap) or conjunctions($\&$), complements or negations ($()'$), material implications ($(\cdot)' \cup (\cdot)$ or $(\cdot)' \vee (\cdot)$, commonly denoted by \Rightarrow , and material double implication or material or logical equivalence ($(\Rightarrow \cdot) \& (\cdot \Rightarrow)$), commonly denoted by \Leftrightarrow . For purposes of simplicity and because the author has propositional logic and its extensions in mind, the common notation used throughout this paper will be the logical/propositional interpretations- but these can all be immediately converted to the set notation counterparts, where required. The corresponding semantic or numerical evaluation for PL is of course simply the choice of a particular (joint) probability measure (either finitely additive or countably additive, if need be) $p: R \rightarrow [0,1]$, the last symbol denoting the unit interval. (See [1] and [4] for background.)

As pointed out in [1] and elsewhere ([4], Chpts 1, 2.2.1), other processes are also involved in establishing an evaluation of a situation such as that entailing C^3 and data fusion processes, besides the choice of ALDP. For example, mental imaging and cognitive processes play an important role, as do natural language formulations, semiotics, and full formal language / formal theory. For simplicity only the latter will be considered in any detail prior to the choice of ALDP. This sequence of knowledge flow in converting any qualitative description into a quantitative one can be summarized in the diagram in Figure 4:



Examples of ALDP's include:

probability logic(PL): (boolean algebra, prob. measure)
fuzzy logic(FL): (boolean lattice, possibility meas.)
 Dempster-Shafer logic(DSL): (boolean algebra, belief ms.)
classical logic (CL): (boolean algebra, 0-1 valued ms.)
(See [1] or [4] again for further details.)

In the case of C^3 processes, an improved full formal language description has been developed for the dynamic evolution of a typical C^3 node state vector [5], replacing previous efforts in [1]-[3]. The next section describes this.

1.3 Full formal language description/ theory for C^3 node evolutions.

In brief, the full formal language description is summarized in the following tables:

EQUALITY SYMBOL: =
CONSTANTS: Ω, \emptyset
DUMMY VARIABLES: α, β, γ
SPECIFIC VARIABLES: N, R, S, T IMPLICIT IN N
OPERATORS: $(\cdot)^+, (\cdot)^-, (\cdot)_0, (\cdot \cdot), \&, \vee, \text{DOM}, \epsilon$
GENERAL AXIOMS: FOR ALL α, β, γ , AND FOR $\ast = \&, \vee$:
RING STRUCTURE FOR $\&, \vee$:
$\alpha \ast \beta = \beta \ast \alpha, \alpha \ast (\beta \ast \gamma) = (\alpha \ast \beta) \ast \gamma,$
$\alpha \& \emptyset = \emptyset, \alpha \& \Omega = \alpha = \alpha \vee \emptyset, \alpha \vee \Omega = \Omega,$
$\alpha \& (\beta \vee \gamma) = (\alpha \& \beta) \vee (\alpha \& \gamma).$

Table 2a. Formal Language Description of a C^3 Node Evolution: Part 1.

GENERAL AXIOMS: FOR ALL α, β, γ , AND FOR $\ast = \&, \vee$:
IMPLICATIVE/CONDITIONAL STRUCTURE FOR $\&, \vee$:
$(\alpha \Omega) = \alpha, (\alpha \beta) = (\alpha \& \beta \beta),$
$(\alpha \& \beta \gamma) = (\alpha \beta \& \gamma) \& (\beta \gamma),$
$(\alpha \ast \beta \gamma) = (\alpha \gamma) \ast (\beta \gamma),$
$\vee_{\alpha \in \text{DOM}(\alpha)} \alpha = \Omega,$
SUFFICIENCY AXIOMS: FOR $N \notin \emptyset$ ($N \& N^- = \& \cdot \& N_0$):
$(R^{++} N^+ \& \bar{N}) = (R^{++} N^+)$
$(N^{++} R^{++} \& N^+ \& \bar{N}) = (N^{++} R^{++} \& N^+)$
$(N^+ S \& R^- \& \bar{N}) = (N^+ S \& R^-)$
$(S R^- \& \bar{N}) = (S R^-)$

Table 2b. Formal Language Description of a Node Evolution: Part 2.

The symbols in Table 2 can be interpreted as follows, as given in Table 3:

N = NODE STATE VECTOR, T = NODE STATE STRUCTURE
R = RESPONSE VECTOR, S = "SIGNAL" VECTOR
$(\cdot)^+$ = POSITIVE TIME SHIFT TO NEW PHASE
$(\cdot)^-$ = NEGATIVE TIME SHIFT TO OLD PHASE
$(\cdot)_0$ = INITIALIZATION OF STATE (TIME-WISE)
$(\cdot \cdot)$ = IMPLICATION OR CONDITIONING
$\&$ = AND, \vee = OR, $(\cdot)'$ = NOT (EXPLAINED EARLIER)
DOM = DOMAIN OF POSSIBLE VALUES
ϵ = SET MEMBERSHIP RELATION AS USED BEFORE

Table 3. Interpretations of the Formal Language for C^3 Node Evolution.

Note that N and T above can be partitioned into subvectors as e.g.:

$$N = (\#WP_1, \#WP_2, \#WP_3, \#TROOP, EQMO, INFO) \quad (1.1)$$

$$T = (\text{DET}, \text{ALG}, \text{HYP}, \text{FUS}, \text{CONS}, \text{DEC}) \quad (1.2)$$

where DET=detection (or not), ALG=algorithm selection, HYP=hypotheses formulation, FUS=data fusion, DEC=decision, etc. One can add the constraint (1.3) to the axioms in Table 2b:

$$(R^{++} | T^+ \& N^+) = (R^{++} | \text{DEC}^+ \& N^+). \quad (1.3)$$

Using similar sufficiency assumptions, implicative chaining in Table 2b shows that

$$\begin{aligned} (T^+ | N^+) &= (\text{DEC}^+ | \text{CONS}^+ \& \text{FUS}^+ \& \text{HYP}^+ \& \text{ALG}^+ \& \text{DET}^+ \& N^+) \\ &\& (\text{CONS}^+ | \text{FUS}^+ \& \text{HYP}^+ \& \text{ALG}^+ \& \text{DET}^+ \& N^+) \\ &\& (\text{FUS}^+ | \text{HYP}^+ \& \text{ALG}^+ \& \text{DET}^+ \& N^+) \\ &\& (\text{HYP}^+ | \text{ALG}^+ \& \text{DET}^+ \& N^+) \\ &\& (\text{ALG}^+ | \text{DET}^+ \& N^+) \& (\text{DET}^+ | N^+). \end{aligned} \quad (1.4)$$

Finally, applying the usual deduction procedure to the axioms given in Table 2, yields the following theorem (1.1) describing the dynamic evolution of a node state in formal language terms:

Theorem 1.1 (See [5].)

Under the assumptions in Table 2:

$$(N^{++} | N) = \left(\begin{matrix} R^{++} \vee \\ \epsilon \text{DOM}(R^{++}) \\ N^+ \epsilon \text{DOM}(N^+) \end{matrix} \right) ((N^{++} | R^{++} \& N^+) \& (R^{++} | N^+) \& (N^+ | N)) \quad (1.5)$$

where $(N^+ | N) = \vee_{R^+ \in \text{DOM}(R^+)} ((N^+ | R^+ \& N) \& (R^+ | N)), \quad (1.6)$

$$(R^{++} | N^+) = \vee_{T^+ \in \text{DOM}(T^+)} ((R^{++} | T^+ \& N^+) \& (T^+ | N^+)), \quad (1.7)$$

$$(N^+ | R^- \& N) = \vee_{S \in \text{DOM}(S)} ((N^+ | S \& N) \& (S | R^-)), \quad (1.8)$$

and $N^{++} = \vee_{N \in \text{DOM}(N)} ((N^{++} | N) \& N). \quad (1.9)$

Thus, using the interpretation in Table 3, compatible with Figure 3,

$$(R^{++}|T^{++}N^{++}) = \text{response following processing,} \quad (1.10)$$

$$(N^{++}|R^{++}N^{++}) = \text{new node state due to its sending out response,} \quad (1.11)$$

$$(T^{++}|N^{++}) = \text{processing data,} \quad (1.12)$$

$$(N^{++}|N) = \text{full cycle of node change due to "signals" received, over all possible processing, and responses,} \quad (1.13)$$

etc.

Thus, if PL were chosen as the ALDP, assuming only stochastic relations are involved in C^3 variables, then Theorem 1 reduces to the more familiar form

$$p(N^{++}|N) = \int_{\substack{\text{over all} \\ R^{++} \in \text{DOM}(R^{++}), \\ N^{++} \in \text{DOM}(N^{++})}} p(N^{++}|R^{++}, N^{++}) \cdot p(R^{++}|N^{++}) \cdot p(N^{++}|N) dR^{++} dN^{++} \quad (1.14)$$

Or, if FL were chosen as the ALDP, assuming only fuzzy relations are involved in C^3 variables, then Theorem 1 becomes under semantic evaluation

$$\text{poss}(N^{++}|N) = \max_{\substack{\text{over all} \\ R^{++} \in \text{DOM}(R^{++}), \\ N^{++} \in \text{DOM}(N^{++})}} (\min(\text{poss}(N^{++}|R^{++}, N^{++}), \text{poss}(R^{++}|N^{++}), \text{poss}(N^{++}|N))), \quad (1.15)$$

One could also choose combinations of PL and FL or other ALDP's in the evaluation aspect. (Again, see [4])

In turn, utilizing the outputs in Theorem 1.1, together with the choice of ALDP, it is clear that the evolution of node states depend on the determination of the relative primitive relations given as sufficiency axioms in Table 2b. Calling each possible combination of such relations for each C^3 side j , friendly ($j=1$) or adversary ($j=2$), PRIM_j , one can establish a loss $L(\text{PRIM}_1, \text{PRIM}_2)$ based directly on Theorem 1.1, thereby establishing a zero sum two-person C^3 decision game and then proceed to analyze the game for values, least favorable strategies, bayes decisions, minimax strategies, etc. (See [5] for further details, where a multidimensional gaussian linear conditional structure is imposed upon the relations yielding feasible computational forms.)

Throughout all of the above calculations, data fusion plays a central role (see again (1.4) and (1.7) where the specific quantitative relations depending upon data fusion are shown). In the next section, motivation is developed for a systematic approach to the fusion proper aspect for disparate information arriving in conditional form.

2. NEED FOR A CONDITIONAL EVENT ALGEBRA IN EVALUATING DATA FUSION

2.1 Introduction.

Noting that many of the relations in Table 2b and subsequent equations are in conditioned form, consider in particular the basic data fusion factor in (1.4)

$$Q \triangleq (FUS^{++}|HYP^{++} \& ALG^{++} \& DET^{++} \& N^{++}), \quad (2.1)$$

noting that the incoming "signal" S is present through the change of N to N^{++} appearing in the antecedent. Fix throughout the discussion, in the antecedent for Q , an arbitrary combination of possible domain values for HYP^{++} , ALG^{++} , DET^{++} , and N^{++} and hence S , up to some variability. Consider then the two following simplified examples:

$$Q = (a|b) \vee (a|c), \quad (2.2)$$

where

- a = ship target possible position area updated,
- b = track history 1 is the assumed assignment (possibly in error) to the target,
- c = track history 2 is the assumed assignment (possibly in error) to the target.

Or, perhaps, Q represents intelligence information to be fused by evaluating the truth of the expression

pression

$$Q = ((a|b) \& (c|d)) \vee (e|f), \quad (2.3)$$

where now

- $a = a(x)$ = enemy will move up about x troops tomorrow; $x=0,50,100,150$.
- $b = b(y)$ = it will y tomorrow; y =be clear, snow, rain.
- $c = c(z)$ = enemy will use Pass z to approach us; $z=I,II,III,IV$.
- $d = d(r,s)$ = morale of enemy node 17 is at level r and number of their troops left is s ; r =very low, low, medium, high, very high, $s=0,100,200,300$.
- $e = e(w)$ = enemy will w tomorrow; w =surrender, not surrender.
- $f = f(q)$ = enemy overall damage level is q ; $q=0,1,2,\dots,10$.

Of course, if the antecedents in (2.2) or (2.3) were all the same, then no real problem would arise, since for example it is readily justified that for any choice of ALDP - certainly for PL - that

$$Q = ((a|d) \& (c|d)) \vee (e|d) = (((a\&c)\vee e)|d), \quad (2.4)$$

even though normally one does not talk about such *measure-free* entities (up to now). Indeed, since the goal is the evaluation of Q , for PL, choosing a probability measure p over all the relevant events, one would usually evaluate Q as simply

$$p(Q) = p(((a|d) \& (c|d)) \vee (e|d)) = p(((a\&c)\vee e)|d) = p(((a\&c)\vee e)\&d)/p(d), \quad (2.5)$$

etc., assuming $p(d)>0$.

But the point of the above examples in (2.2) and (2.3) is that the antecedents in the conditional forms are *not* in general identical! What to do?

Contrary to popular belief [author's note: this author and his colleague Prof. H.T. Nguyen, Math. Dept. New Mexico State University, Las Cruces, have undertaken an extensive informal survey of the probability community- both applied and theoretical- resulting in the following conclusions - see also [7]]; there is no systematic and mathematically sound procedure for computing $p(Q)$ (or Q , for that matter) in either (2.2) or (2.3), or any similar problem!

Indeed, there do exist "folk" remedies to this situation which roughly speaking reduce to two approaches as presented in the next sections.

2.2 Approach A: Identification of conditioning with material implication.

In this approach, one interprets $(a|b)$ as $b \Rightarrow a$, i.e., for any a, b , events,

$$(a|b) = b \Rightarrow a \triangleq b' \vee a \triangleq b' \vee (a\&b) \quad (2.6)$$

is assumed to be valid so that by the principle of substitution relative to equality, for any (suitable) probability measure p , assuming the identity for $p(b)>0$

$$p((a|b)) = p(a|b) \triangleq p(a\&b)/p(b), \quad (2.7)$$

one has immediately from (2.6)

$$p(a|b) = p(b \Rightarrow a) = p(b' \vee a) = 1 - p(b') + p(a\&b), \quad (2.8)$$

etc.

Thus using this approach, (2.2) becomes:

$$p(Q) = p((b \Rightarrow a) \vee (c \Rightarrow a)) = p(b' \vee a \vee c' \vee a) = p((b\&c)' \vee a) = p(a|b\&c). \quad (2.9)$$

However, there is just one flaw in the above reasoning: One cannot make the identification in general in the left hand side of (2.6). Indeed, it is rather easy to show (yet surprisingly few are aware of the inequality below- see e.g. the discussion in [7], sect. 1.8):

$$p(b \Rightarrow a) = p(a|b) + p(a'|b) \cdot p(b') \geq p(a|b) \geq p(a\&b), \quad (2.10)$$

where in general strict inequality holds for the last two \geq . Furthermore, (2.6) is not even a good approximation, since it is readily verified for b with $p(b)$ small that one can choose a and b such that $p(b \Rightarrow a)$ is close to one, while $p(a|b)$ is close to zero. This unfortunate situation is a special case of the "Stalnaker Thesis" problem and is considered in detail in [6] and [7], section 1. In fact the following apparently not-well-known result has been shown:

Theorem 2.1. (P. Calabrese [8], section 1.2)

Let R be a boolean algebra of events with the usual operations discussed previously. Let p be any non-degenerate probability measure over R , i.e. $p: R \rightarrow [0,1]$. Then, there is no binary boolean function $f: R^2 \rightarrow R$ such that for all $a, b \in R$, with $p(b) > 0$,

$$p(a|b) = p(f(a,b)). \quad (2.11)$$

The result is further extended in one direction by the following result:

Theorem 2.2. (Goodman & Nguyen [7], Theorem 2.7)

Let R be any finite boolean algebra of events. Then, there is no binary function f of any kind, $f: R^2 \rightarrow R$, such that for all $a, b \in R$, with $p(b) > 0$, (2.11) holds.

However, by allowing infinite boolean algebras one can force a form of conditioning to be back in the original boolean algebra (but not without complication as Copeland showed. (For a critique of Copeland's "implicative" boolean algebras, see [7], section 1.8.)

2.3 Approach B: Identification of conditional events as marginal ones with common joint antecedents.

In this approach, one attempts to obtain a common joint antecedent event for all of the conditional events appearing and then identify each as a marginal one having a common joint antecedent. In turn, one proceeds to evaluate for a suitably chosen probability measure analogous to that in (2.5). This is best shown through an application to (2.3):

First make the identifications

$$\begin{aligned} & \{a|b \text{ with } (a \times d \times f|G)\} \\ & \{c|d \text{ with } (b \times c \times f|G)\} ; G \stackrel{d}{=} b \times d \times f. \end{aligned} \quad (2.12)$$

Then (2.3) becomes

$$\begin{aligned} Q &= ((a \times d \times f|G) \& (b \times c \times f|G)) \vee (b \times d \times e|G) \\ &= (((a \& b \times c \& d \times f) \vee (b \times d \times e))|G), \end{aligned} \quad (2.13)$$

resulting in the evaluation

$$p(Q) = p((a \& b \times c \& d \times f) \vee (b \times d \times e) | G), \quad (2.14)$$

which can be further reduced using the definition of conditional probability and the standard calculus of operations for PL.

However, the main drawback to this approach is that initial probability measure $p: R \rightarrow [0,1]$ must be replaced by some joint probability measure p_0 over the boolean (or sigma-) algebra spanned by R^3 , where $p(\cdot|b), p(\cdot|d), p(\cdot|f): R \rightarrow [0,1]$ are conditional probability measures whose joint measure is p_0 , so that

$$p(a|b) = p_0(a \times d \times f|G), p(c|d) = p_0(b \times c \times f|G), p(e|f) = p_0(b \times d \times e|G) \quad (2.15)$$

But what choice of p_0 to make? Should it be based on maximal entropy considerations, etc.? Long calculations can also result from the cartesian product forms. (For further discussion, see [7], sections 1.1, 1.5, and 10.4. However, for a tie-in with the approach presented in this paper, Theorem 4.2 given in section 4 is of use.)

2.4 The basic problem in representing conditional events with distinct antecedents.

As shown in section 2.3 two of the most common approaches to the handling of conditional events do not lead to satisfactory results, from either a mathematical viewpoint, as in Approach A, or an unambiguous computationally efficient viewpoint, as in Approach B.

Thus one is lead to question whether any remedy exists for this situation: Can a calculus of measure-free conditional events be developed which is both compatible with ordinary conditional probability evaluations and is also unambiguous and feasible to implement, as well as being based upon sound, non-ad hoc mathematical principles? Certainly, all evaluation of fused data, and hence evaluations for the overall C^3 problem, must depend, in effect, on the outcome of this question.

It is the contention here that the answer to the above question is definitely in the affirmative. In this paper in section 3 an outline of a theory for developing such a calculus of operations and related issues is presented. This is carried out for not only the purpose of keeping this paper as self-contained as possible, but also because of the desire to disseminate these novel and universally applicable results to as wide an audience as possible, within a short time period. For earlier efforts in this direction, see [6]. In [7] (in the process of being submitted for publication) the full theory, with all proofs, is exhibited.

One consequence of the calculus of conditional events is that the evaluations for Q in (2.2) and (2.3) become rather simple. Thus, it will be shown that (2.2) yields

$$Q = ((a \& (b \vee c)) | (a \& (b \vee c)) \vee (b \& c)) \quad (2.16)$$

so that

$$p(Q) = p(a \& (b \vee c)) / p((a \& (b \vee c)) \vee (b \& c)), \quad (2.17)$$

etc., differing considerably from that proposed by Approaches A (see (2.9)) or B.

For (2.3), one obtains

$$Q = (a|\beta) \quad (2.18)$$

resulting in

$$p(Q) = p(a) / p(\beta), \quad (2.19)$$

where

$$\alpha \stackrel{d}{=} (a \& b \& c \& d) \vee (e \& f), \quad (2.20)$$

$$\beta \stackrel{d}{=} \alpha \vee ((a' \& b) \vee (c' \& d) \vee (b \& d) \& f), \quad (2.21)$$

differing again considerably from both Approaches A and B (see (2.13)).

In all of the above computations, one need not construct joint probability measures and consider cartesian products of events, nor is the procedure ad hoc (despite the oversimplified appeal of Approach A - but see section 4(9)).

3. OUTLINE OF A THEORY OF MEASURE-FREE CONDITIONAL EVENTS

3.1 Introduction.

Following the basic motivation for the development of a conditional event calculus for PL in section 2, this section presents an overview of the basic results. The following three questions are addressed:

(i) What meaning can be attached to a typical conditional event $(a|b)$, where a and b are ordinary unconditional events, prior to evaluating through a specific probability measure p to become (for $p(b) > 0$)

$$p((a|b)) = p(a|b) \stackrel{d}{=} p(a \& b) / p(b) ? \quad (3.1)$$

(ii) How shall operations - in particular, the usual boolean operations $\&, \vee, ()'$ and relations such as \leq - be extended from unconditional events to conditional ones and what properties do they possess?

(iii) Can such operations and relations as in (ii) be characterized for uniqueness, etc. ?

For a history of previous attempts at developing a theory of measure-free conditioning, see [7], sections 1.8 and 1.9. Among the predecessors of this effort, Schay [9] was among the first to attempt such a task, but used an ad hoc procedure in addressing question (ii), although a somewhat complicated characterization was developed relative to (iii). Later, Calabrese, completely independent of Schay, produced also an algebra of conditional events and operations,

the latter also from an ad hoc approach following certain analogies with material implication [8].

3.2 Development of measure-free conditional events.

From now on, without further explanation, the symbol R refers to an arbitrary fixed boolean algebra with all of the usual operations and relations explained previously (see section 1.2). The partial order \leq over R^2 (corresponding to the subset relation \subseteq among sets corresponding to events or propositions as considered here) is defined as usual as

$a \leq b$ iff $a = a \& b$ iff $b = a \vee b$, (3.2)
for any (unconditional) events $a, b \in R$. Note also, \emptyset denotes the null event and Ω the universal (unity) event.

Define (f, S) to be any candidate class of conditional events extending R iff f is some function $f: R^2 \rightarrow S$ for some space S , such that for all $a, b, c \in R$, and boolean operations $*$ over R ; $*$, etc. are operations over (assumed) boolean algebra of events $S_b \triangleq f(\cdot, b)(R) = \text{range}(f(\cdot, b))$ (3.3)

where

$$S = \bigcup_{b \in R} S_b, \quad (3.4)$$

$$f(a, b) * f(c, b) = f(a * c, b), \quad (3.5)$$

and

$$f(a, b) = f(a \& b, b), \quad f(a, \Omega) = a. \quad (3.6)$$

For any candidate class (f, S) , one can interpret $f(a, b)$ as a conditioned upon b wrt f , or symbolically as $(a|b)_f$, where a is the consequent and b the antecedent.

Theorem 3.1, ([7], Theorem 2.1)

(i) (nat, \bar{R}) is a candidate class of conditional events extending R , called the canonical class of conditional events extending R , where in (3.4)

$$R_b \triangleq \{(a|b)_{\text{nat}} : a \in R\}, \quad (3.7)$$

where for each $a, b \in R$,

$$(a|b) \triangleq (a|b)_{\text{nat}} \triangleq \{(R \& b') \vee (a \& b) : \{ (x \& b') \vee (a \& b) : x \in R \} \} \quad (3.8)$$

is the principal ideal coset ($R \& b'$ being the principal ideal) generated by b' with residue $a \& b$, making R_b a boolean quotient algebra with the usual coset operations $*$ over R (thus, $*$ = $\&, \vee, ()', \text{etc.}$).

$$\bar{R} \triangleq \bigcup_{b \in R} R_b = \{(a|b) : a, b \in R\} \quad (3.9)$$

and for any $a, b \in R$,

$$\text{nat}(a, b) \triangleq (a|b). \quad (3.10)$$

(ii) For each $a, b \in R$, noting $(a|b) \in R$, $(a|b)$ is the inverse of the conjunction operation $(\&b): R \rightarrow R$ at $a \& b$, i.e.,

$$(a|b) = \{y : y \in R \text{ and } y \& b = a \& b\}. \quad (3.11)$$

Equivalently, $(a|b)$ is the largest subclass of R satisfying the relation

$$(a|b) \& b = a \& b \quad (3.12)$$

(iii) For any candidate class (f, S) of conditional events extending R , and each $b \in R$,

$$S_b(\&b, \vee_b, ()') \approx_b R/f^{-1}(\cdot, b) \text{ (boolean quot. alg.)} \\ \geq R_b(\&b, \vee_b, ()'), \quad (3.13)$$

the symbol \approx denoting an isomorphism, so that in the above local sense for each b , the canonical class of conditional events is the smallest possible candidate class of conditional events extending R .

Note also that for any $a, b, c, d \in R$,

$$(a|b) = (c|d) \text{ iff } a \& b = c \& d \text{ and } b = d, \quad (3.14)$$

but unlike the classical case where $b = d$, $(a|b)$ is not necessarily identical nor disjoint from $(c|d)$. (See [7], Theorem 2.11.) Note also the relations

$$R \subseteq \bar{R} \subseteq P(R), \quad (3.15)$$

where $P(R)$ denotes the power class, or class of all subsets of elements, of R .

Next, call any pair (f, S) (not necessarily a priori a candidate class of conditional events) where $f: R^2 \rightarrow S$ is surjective (i.e., onto) a probability-compatible pair with respect to R iff, by definition,

$$R \subseteq S = \bigcup_{b \in R} S_b \quad (3.16)$$

where now S_b is some boolean algebra, for each $b \in R$, and where for each probability measure $p: R \rightarrow [0, 1]$ and each $b \in R$ with $p(b) > 0$, p can be extended to (using the same symbol) $p: S \rightarrow [0, 1]$ such that for all $a \in R$,

$$p(f(a, b)) = p(a|b). \quad (3.17)$$

Theorem 3.2, ([7], Theorems 2.5, 2.9)

Let (f, S) be a probability-compatible pair with respect R . Then:

(i) (f, S) is also a candidate class of conditional events extending R .

(ii) (nat, \bar{R}) is a probability-compatible pair.

(iii) By suitable restriction of $f: R^2 \rightarrow S$ to non-trivial, i.e., non-zero or non-unity probability-valued conditional event pairs, then f becomes bijective and \approx_b in (3.13) can be replaced by a global isomorphism \approx not dependent upon b , showing that the canonical class of conditional events (nat, \bar{R}) is the universally minimal (wrt subclass inclusion) probability-compatible candidate class of conditional events extending R .

Next, a basic logical justification for choosing the canonical conditional events is given. First, for any $a, b \in R$, define the class of all a -relative deducts of b as

$$\{a|b\} \triangleq \{x : x \in R \text{ and there exists } r \in R \text{ with } r \& b \text{ and } x \& r = a \& r\} \quad (3.18)$$

(Calabrese [8], motivated by classical logic deduction, proposed this definition previously for conditioning a by b .)

Theorem 3.3, ([7], Theorem 1.3)

For all $a, b \in R$,

$$\{a|b\} = (a|b), \quad (3.19)$$

i.e., \bar{R} coincides with the class of all relative deducts!

3.3 Calculus of operations on conditional events.

With the role of the canonical class of conditional events firmly established in section 3.2, consider now the choice of operations upon them extending the usual boolean operations acting upon unconditional events. Throughout all of real analysis and topology, a universal way of extending a given "point"-valued function to a "set"-valued one is simply through the natural extension or, equivalently, called the component-wise class or image extension. In particular, let $g: R^2 \rightarrow R$ be any binary operation (boolean or otherwise) upon R (unary and more generally, n -ary operations can be treated similarly). Then the natural extension (also denoted by the same symbol for g) from g over R^2 to $g: P(R)^2 \rightarrow P(R)$, i.e., over $P(R)^2$, is determined by,

$$g(A, B) \triangleq \{g(a, b) : a \in A, b \in B\}, \quad (3.20)$$

for any $A, B \subseteq R$, i.e., $A, B \in P(R)$.

Thus, recalling the comments in Theorem 3.1(ii) and (3.15), it is basic to inquire what forms the natural extensions take for the ordinary boolean operations relative to R when restricted to \bar{R} . Note that for the binary operations $\&, \vee: R^2 \rightarrow R$, by their commutativity and associativity; they extend recursively

and unambiguously to $\&, v: R^n \rightarrow R$, where, by convention, for $n=1$, $\& = v =$ identity function. (The unary negation or complement operator $()': R \rightarrow R$ remains as is.) Denote the natural extensions of $\&, v, ()'$ by the same symbols. Thus, here the behavior of the natural extensions of $\&, v: R^n \rightarrow R$ to the restrictions $\&, v: \tilde{R}^n \rightarrow P(R)$ and that of $()': R \rightarrow R$ to the restriction $()': \tilde{R} \rightarrow P(R)$ are sought.

Theorem 3.4. ([7], Corollaries 3.3, 3.4)

For any positive integer n and any $a_j, b_j \in R$, $j=1, \dots, n$:

(i) The common antecedent case, noting $R_b \subseteq R$.

$$(a|b)' = (a|b)'_b = (a'|b) = (a' \& b|b), \quad (3.21)$$

$$\bigwedge_{j=1}^n (a_j|b) = \bigwedge_{j=1}^n (a_j|b)_b = \left(\bigwedge_{j=1}^n a_j | b \right) = \left(\bigwedge_{j=1}^n a_j \& b | b \right), \quad (3.22)$$

for $\ast = \&, v$.

Thus, relative to any given boolean quotient algebra of the form R_b , all coset operations and corresponding natural extensions of the original operations relative to R coincide. It follows that if $g: R^n \rightarrow R$ is any (compound) boolean operation, then

$$g((a_1|b_1), \dots, (a_n|b_n)) = \tilde{g}_b(a_1|b_1), \dots, (a_n|b_n)) \\ = (g(a_1, \dots, a_n)|b), \quad (3.23)$$

(ii) Distinct antecedents, in general, case.

$$\bigwedge_{j=1}^n (a_j|b_j) = \left(\bigwedge_{j=1}^n a_j | r \right) = \left(\bigwedge_{j=1}^n (a_j \& b_j) | r \right), \quad (3.24)$$

$$\text{where } r = \bigwedge_{j=1}^n b_j \vee \bigvee_{j=1}^n v(a_j' \& b_j) = \bigwedge_{j=1}^n (a_j \& b_j) \vee \bigvee_{j=1}^n v(a_j' \& b_j) \\ \text{and} \quad (3.25)$$

$$\bigvee_{j=1}^n v(a_j|b_j) = \left(\bigvee_{j=1}^n a_j | q \right) = \left(\bigvee_{j=1}^n (a_j \& b_j) | q \right), \quad (3.26)$$

$$\text{where } q = \bigwedge_{j=1}^n v(a_j \& b_j) \vee \bigwedge_{j=1}^n b_j = \bigvee_{j=1}^n v(a_j \& b_j) \vee \bigwedge_{j=1}^n v(a_j' \& b_j). \quad (3.27)$$

Combining (3.24) and (3.26), leads to the following corollary:

Corollary 3.1 (New result.)

Noting that any (compound) boolean function of multiple arguments can always be put into an equivalent form consisting of a disjunction of conjunctions, let $a_{ij}, b_{ij} \in R, i=1, \dots, m, j=1, \dots, n$, where some of these events may be \emptyset or Ω . Then

$$\bigvee_{i=1}^m \bigwedge_{j=1}^n (a_{ij}|b_{ij}) = (\alpha | \beta), \quad (3.28)$$

$$\text{where } \alpha = \bigvee_{i=1}^m \bigwedge_{j=1}^n v(a_{ij} \& b_{ij}), \quad \beta = \alpha + \gamma, \quad (3.29)$$

$$\gamma = \bigwedge_{j=1}^n \bigvee_{i=1}^m v(a_{ij}' \& b_{ij}), \quad (3.30)$$

noting that α and γ are disjoint, i.e.,

$$\alpha \& \gamma = \emptyset, \quad (3.31)$$

so that for any probability measure $p: R \rightarrow [0, 1]$,

$$p\left(\bigvee_{i=1}^m \bigwedge_{j=1}^n (a_{ij}|b_{ij})\right) = p(\alpha|\beta) = p(\alpha)/p(\beta) = 1/(1 + (p(\gamma)/p(\alpha))), \quad (3.32)$$

Theorem 3.6. Some special cases ([7], section 4.2).

For any $a, b, c, d, a_i \in R, i=1, \dots, n$:

$$(a|\Omega) = a, (a|\emptyset) = R, (\emptyset|b) = (b'|b) = R \& b' = \{x \& b' : x \in R\}, \quad (3.33)$$

$$(\Omega|b) = (b|b) = R \vee b' = \{x \vee b' : x \in R\}, \quad (3.34)$$

$$(a|b) \& (c|d) = (a \& b \& c \& d | (a' \& b) \vee (c' \& d) \vee (b \& d))$$

$$= (a \& b \& c \& d | (a' \& b) \vee (c' \& d) \vee (a \& b \& c \& d)), \quad (3.35)$$

$$(a|b) \vee (c|d) = ((a \& b) \vee (c \& d) | (a \& b) \vee (c \& d) \vee (b \& d)) \\ = ((a \& b) \vee (c \& d) | (a \& b) \vee (c \& d) \vee (a' \& b \& c' \& d)), \quad (3.36)$$

$$(a|b) = (a \& b|b) = (b \Rightarrow a|b) = (b \Rightarrow a) \& (b|b) = ((b'|a') \vee a) \& (b|b), \quad (3.37)$$

$$(a|b) \& b = a \& b, (a|b) \vee b = (b|b), \quad (3.38)$$

$$(a|b) \vee (b|a) = (a \& b|a \& b), (a|b) \& (b|a) = (a \& b|a \vee b), \quad (3.39)$$

$$(a|b) \vee (c|b') = ((a \& b) \vee (c \& b')) | (a \& b) \vee (c \& b'), \quad (3.40)$$

$$(a|b) \& (c|b') = (\emptyset | (a' \& b) \vee (c' \& b')), \quad (3.41)$$

$$(c|d) \Rightarrow (a|b) = (c \Rightarrow a | (c' \& d) \vee (a \& b) \vee (b \& d)), \quad (3.42)$$

$$(c|d) \Leftrightarrow (a|b) = (c \Leftrightarrow a | b \& d), \quad (3.43)$$

$$\text{chaining: } \begin{cases} (a \& b|c) = (a|b \& c) \& (b|c), \\ (a \& b \& c|d) = (a|b \& c \& d) \& (b|c \& d) \& (c|d), \end{cases} \quad (3.44)$$

$$\text{Bayes' Thm: } (a_i|b) = ((b|a_i) \& a_i | \bigvee_{j=1}^n ((b|a_j) \& a_j)), \quad (3.46)$$

$$\text{provided that } \begin{matrix} n \\ b \leq v a_j, \\ j=1 \end{matrix} \quad (3.47)$$

where above, material implication and material double implication, as extended from R to \tilde{R} , are defined formally the same as in the unconditional case:

$$(c|d) \Rightarrow (a|b) \stackrel{d}{=} (c|d)' \vee (a|b), \quad (3.48)$$

$$(c|d) \Leftrightarrow (a|b) \stackrel{d}{=} ((c|d) \Rightarrow (a|b)) \& ((a|b) \Rightarrow (c|d)), \quad (3.49)$$

Recalling the partial order \leq for R (see (3.2)), define and extend this relation to \tilde{R} by letting, for all $a, b, c, d \in R$,

$$(a|b) \leq (c|d) \text{ iff } (a|b) = (a|b) \& (c|d). \quad (3.50)$$

Theorem 3.7. ([7], Theorems 4.1, 4.7)

For all $a, b, c, d \in R$:

(i) Characterization of \leq over \tilde{R}^2 :

$$(a|b) \leq (c|d) \text{ iff } (c|d) = (a|b) \vee (c|d) \\ \text{iff } (a \& b) \leq (c \& d) \text{ and } (b \Rightarrow a) \leq (d \Rightarrow c) \\ \text{iff } (a \& b) \leq (c \& d) \text{ and } (c' \& d) \leq (a' \& b) \\ \text{iff } (c|d)' \leq (a|b)'. \quad (3.51)$$

(ii) \leq over \tilde{R}^2 is not only a partial order (reflexive, transitive, anti-symmetric), but also a meet ($\&$) and join (\vee) lattice with all of the usual operation-preserving properties. Note the relation, compatible with (2.10) and (3.8), showing that $b \Rightarrow a$ and $a \& b$ are the largest and smallest elements in $(a|b)$, respectively, relative to partial order \leq :

$$a \& b \leq (a|b) \leq b \Rightarrow a. \quad (3.52)$$

(iii) $\tilde{R}(\&, v, ()') \leq$ is a distributive lattice which is also an algebraic semi-ring with zero element \emptyset and unity element Ω , and is, further, idempotent, absorbent, and demorgan, among other properties.

3.4 Justifications for choice of extensions of operations from R to \tilde{R} .

In addition to the large number of desirable properties for the naturally extended boolean operations upon \tilde{R} given in section 3.3, characterizations can also be established.

Recalling the maximality property of $b \Rightarrow a$ with respect to $(a|b)$ ((3.52)), define a corresponding mapping $\phi: \tilde{R} \rightarrow R$, where for any $a, b \in R$,

$$\phi((a|b)) \stackrel{d}{=} b \Rightarrow a. \quad (3.53)$$

Theorem 3.8. ([7], Theorem 10.1, Remark 10.1)

(i) ϕ is a surjective $\&, v$ -homomorphism with respect to the natural extensions of $\&, v$ from R to \tilde{R} .

(ii) Let $\tilde{\&}, \tilde{v}: \tilde{R}^2 \rightarrow \tilde{R}$ be any possible extensions of $\&, v: R^2 \rightarrow R$, respectively, such that there exist

functions $\psi_1, \psi_2: R^4 \rightarrow R$, where for all $a, b, c, d \in R$,
 $a \& b \vee c \& d \leq \psi_2(a, b, c, d)$, $a \& b \& c \& d \leq \psi_1(a, b, c, d)$ (3.54)

and

$$(a|b) \& (c|d) = (a \& b \& c \& d | \psi_1(a, b, c, d)), \quad (3.55)$$

$$(a|b) \vee (c|d) = ((a \& b) \vee (c \& d) | \psi_2(a, b, c, d)), \quad (3.56)$$

noting the essential necessity of consequences matching the corresponding ones for the natural extensions.

Then it follows that relative to \tilde{R} ,

$$\tilde{\&} = \&, \quad \tilde{\vee} = \vee. \quad (3.57)$$

(iii) Compatible with Theorems 2.2, 3.1, 3.2, there is no full (i.e., $\&, \vee, ()$) homomorphism $\rho: \tilde{R} \rightarrow R$ with respect to the natural extensions of $\&, \vee, ()$ from R to \tilde{R} .

Theorem 3.9. Partial converse of Theorem 3.8.
 ([7], Theorem 10.2, Corollary 10.1, Remark 10.2)

Let $\tilde{\&}, \tilde{\vee}: R^2 \rightarrow \tilde{R}$ be any possible extensions of the corresponding coset operations $\&, \vee: R^2 \rightarrow R$, simultaneously and consistently for all $b \in R$. (Hence, necessarily, $\tilde{\&}, \tilde{\vee}$ extend $\&, \vee$ relative to R .)

Suppose also that $\tilde{\&}, \tilde{\vee}$ obey not only closure, but are also commutative and associative over \tilde{R} with \emptyset and $\mathbf{1}$ playing the usual roles of zero and unity elements, respectively. Suppose also there exists $\psi_1, \psi_2: R^4 \rightarrow R$ such that for all $a, b, c, d \in R$,

$$(a|b) \& (c|d) = (a \& b \& c \& d | \psi_1(a, b, c, d)), \quad (3.58)$$

$$(a|b) \vee (c|d) = ((a \& b) \vee (c \& d) | \psi_2(a, b, c, d)), \quad (3.59)$$

$$(a|b) \& b = a \& b, \quad (3.60)$$

and

$$((a|b) \& c \in R) \text{ implies } c \leq b. \quad (3.61)$$

Then

(i) For all $a, b, c, d \in R$,

$$(a' \& b) \vee (c' \& d) \vee (b \& d) \leq \psi_1(a, b, c, d), \quad (3.62)$$

$$(a \& b) \vee (c \& d) \vee (b \& d) \leq \psi_2(a, b, c, d), \quad (3.63)$$

showing that the natural extensions of $\&, \vee$ are maximal, i.e.,

$$(a|b) \& (c|d) \leq (a|b) \& (c|d), \quad (3.64)$$

$$(a|b) \vee (c|d) \leq (a|b) \vee (c|d). \quad (3.65)$$

(ii) Result (i) shows that in contradistinction to the antecedent-only dependent operations of Schay [9] and Calabrese [8], there are no boolean functions ψ_1, ψ_2 as above, but now such that for all $a, b, c, d \in R$, $\psi_j(a, b, c, d) = \psi_j(b, d)$ only, $j=1, 2$, such that the corresponding operations $\tilde{\&}$ and $\tilde{\vee}$ satisfy the hypotheses of this theorem.

3.5 Additional properties of \tilde{R} .

Finally, brief mention should be made of other pertinent properties of the conditional extension of a given boolean algebra of events.

(i) Stone's Representation Theorem - showing an order preserving isomorphism always exists between a given boolean algebra R of events or propositions and a corresponding boolean algebra of subsets of some set - can be extended quite readily to \tilde{R} ([7], Theorem 10.3).

(ii) The usual Hilbert-Ackermann axioms involving material implication, relative to any R , when R is replaced by \tilde{R} , ordinary substitution, and a modified form of modus ponens used to deduce theorems from previous theorems and axioms, together with the natural extensions of all boolean operations from R to \tilde{R} , forms a sound and complete logic, i.e., all theorems are tautologies and vice-versa. Call this conditional probability logic (CPL), extending ordinary probability logic (PL). (See [7], Corollary 9.1.) Here, for any $a, b \in R$, $(a|b)$ is a tautology for CPL, written $\models_{\text{CPL}} (a|b)$, iff, by definition,

$$p(a|b) = 1, \quad (3.66)$$

for all probability measures $p: R \rightarrow [0, 1]$, with $p(b) > 0$.

From [7], sections 9.4, 9.5, 2.3, the following concepts and results hold, for any $a, b, c, d \in R$:

$$(I) \models_{\text{CPL}} (a|b) \text{ iff } b \leq a \text{ iff } b \Rightarrow a = 1 \text{ iff } (a|b) = (b|b), \quad (3.67)$$

$$(II) \models_{\text{CPL}} (a|b) \text{ and } \models_{\text{CPL}} (b|a) \text{ iff } a = b \text{ iff } a \Leftrightarrow b = 1 \text{ iff } (a|b) = (a|a) = (b|b), \quad (3.68)$$

$$(III) \text{ Define } (c|d) \text{ tautologically implies } (a|b) \text{ iff } \models_{\text{CPL}} ((c|d) \Rightarrow (a|b)). \quad (3.69)$$

$$\text{Then } \models_{\text{CPL}} ((c|d) \Rightarrow (a|b)) \text{ iff } b \& c \& d \leq a \text{ iff } c \& d \leq b \Rightarrow a. \quad (3.70)$$

$$(IV) \text{ Define } (c|d) \text{ tautologically is equivalent to } (a|b) \text{ iff } \models_{\text{CPL}} ((c|d) \Leftrightarrow (a|b)). \quad (3.71)$$

$$\text{Then } \models_{\text{CPL}} ((c|d) \Leftrightarrow (a|b)) \text{ iff } b \& d \leq (a \& b \Leftrightarrow c \& d). \quad (3.72)$$

$$(V) \models_{\text{CPL}} (c|d) \leq (a|b) \text{ implies } \models_{\text{CPL}} ((c|d) \Rightarrow (a|b)). \quad (3.73)$$

$$(VI) (c|d) = (a|b) \text{ implies } \models_{\text{CPL}} ((c|d) \Leftrightarrow (a|b)). \quad (3.74)$$

$$(VII) \text{ Sufficient conditions for modus ponens analogue: } (c|d) \leq (a|b) \text{ and } \models_{\text{CPL}} (c|d) \text{ implies } \models_{\text{CPL}} (a|b). \quad (3.75)$$

$$(VIII) \text{ Characterization of modus ponens analogue: } \models_{\text{CPL}} ((c|d) \Rightarrow (a|b)) \text{ and } \models_{\text{CPL}} (c|d) \text{ iff } \models_{\text{CPL}} (a \& c | b \& d) \text{ and } \models_{\text{CPL}} (c|d). \quad (3.76)$$

$$(IX) \models_{\text{CPL}} ((a|b) \Leftrightarrow (c|d)) \text{ implies } (\models_{\text{CPL}} (a|b) \Rightarrow (c|d)) \text{ and } \models_{\text{CPL}} ((c|d) \Rightarrow (a|b)). \quad (3.77)$$

$$(X) \text{ Next, define } (c|d) \text{ semantically (or uniformly in probability) implies } (a|b), \text{ written } (c|d) \Rightarrow (a|b), \text{ iff } p(c|d) \leq p(a|b), \quad (3.78)$$

for all probability measures $p: R \rightarrow [0, 1]$ with $p(b), p(d) > 0$.

Also, define any $(a|b)$ to be a contradiction for CPL, written $\nmodels_{\text{CPL}} (a|b)$, iff for all probability

$$\text{measures } p: R \rightarrow [0, 1], \text{ with } p(b) > 0, \quad p(a|b) = 0. \quad (3.79)$$

It readily follows that

$$\nmodels_{\text{CPL}} (a|b) \text{ iff } a \& b = 0 \text{ iff } \models_{\text{CPL}} (a|b)'. \quad (3.80)$$

With a bit more difficulty one can show

$$(c|d) \rightarrow (a|b) \text{ iff } (\models_{\text{CPL}} (a|b) \text{ or } \nmodels_{\text{CPL}} (c|d) \text{ or } (c|d) \leq (a|b)). \quad (3.81)$$

$$(XI) \text{ Call } (a|b) \text{ and } (c|d) \text{ semantically (or uniformly in probability) equivalent, written } (c|d) \Leftrightarrow (a|b), \text{ iff } p(a|b) = p(c|d), \quad (3.82)$$

for all probability measures $p: R \rightarrow [0, 1]$ with $p(b), p(d) > 0$.

It follows that

$$(c|d) \Leftrightarrow (a|b) \text{ iff } (\models_{\text{CPL}} (a|b) \text{ and } \models_{\text{CPL}} (c|d)) \text{ or } (\nmodels_{\text{CPL}} (a|b) \text{ and } \nmodels_{\text{CPL}} (c|d)) \text{ or } (a|b) = (c|d). \quad (3.83)$$

(XII) Finally, mention can be made of a weaker form of semantic implication by restricting probabilities in the above definitions to subclasses or even to a

single probability measure.

(iii) Another area of basic interest concerns higher order conditioning. Thus, if beginning with unconditional events say $a, b, c, d \in R$, it is meaningful to consider the conditional events $(a|b)$, $(c|d)$, then why should one not consider, in turn, the second order conditional event $((a|b)|(c|d))$? (This issue is addressed in [7], section 5 in some detail.) Basically, the appropriate definition for the above expression is the formal higher order analogue of (3.11). Thus

$$((a|b)|(c|d)) = \{(x|y) : (x|y) \in \tilde{R}, (x|y) \& (c|d) = (a|b) \& (c|d)\} \quad (3.84)$$

One major result ([7], Theorem 5.5) concerns the class union mapping $u: P(P(R)) \rightarrow P(R)$, where for any $Q \subseteq P(R)$,

$$u(Q) \triangleq \{x : x \in A \in Q, \text{ for any } A \in Q\}. \quad (3.85)$$

Noting that u is a surjective homomorphism relative to all natural extensions of operations from $P(R)$ to $P(P(R))$, not just boolean operations, and noting that all boolean operations over $\tilde{R} (\subseteq P(R))$ can be extended to $P(\tilde{R}) (\subseteq P(P(R)))$ in the natural sense, though unlike the first order case, \tilde{R} , closure problems arise for the higher order case, the following reduction holds:

$$u(((a|b)|(c|d))) = (a|b \& ((a' \& d') \vee (c \& d))), \quad (3.86)$$

for all $a, b, c, d \in R$. In particular,

$$u(((a|b)|c)) = u(((a|b)|(c|b))) = (a|b \& c). \quad (3.87)$$

Thus, through the above relations, in a sense, one need never consider higher order conditioning.

(iv) One can establish optimal approximations of arbitrary subclasses of R through conditional events, i.e., if $A \subseteq R$, the best upper approximation of A through \tilde{R} satisfies the relation ([7], section 6)

$$A \subseteq \text{cond}(A) \triangleq \bigcap \{(a|b) : A \subseteq (a|b) \in \tilde{R}\} \subseteq \tilde{R}, \quad (3.88)$$

noting trivially that for any $(a|b) \in \tilde{R}$,

$$(a|b) = \text{cond}((a|b)). \quad (3.89)$$

The chief results include the following (see [7], Theorems 6.1, 6.2, and Corollary 6.3):

(I) If now R is a complete boolean algebra, then for any $A \subseteq R$,

$$\text{cond}(A) = (\&(A)|\&(A) \vee (\vee(A))') \in \tilde{R}, \quad (3.90)$$

where

$$\&(A) \triangleq \&a, \quad \vee(A) \triangleq \vee a \quad \begin{matrix} a \in A \\ a \in A \end{matrix} \quad (3.91)$$

are the (possibly non-finite) extended definitions of $\&$ and \vee due to the completeness of R .

(II) Let $R = \mathcal{B}$, the class of all borel subsets of \mathbb{R} , the real (or one-dimensional euclidean) line. Let $g: \mathbb{R}^m \rightarrow \mathbb{R}$ be any function. Denote the natural extension of g to $P(\mathbb{R})$ restricted to $\mathcal{B} (\subseteq P(\mathbb{R}))$ as simply $g: \mathcal{B}^m \rightarrow \mathcal{B}$ (assuming g is sufficiently measurable). In turn, denote the natural extension to $P(\mathcal{B})$ restricted to $\tilde{\mathcal{B}}$, the conditional extension of \mathcal{B} , as simply $g: \tilde{\mathcal{B}}^m \rightarrow P(\mathcal{B})$. Then for all (borel sets) $a_i, b_i \in \mathcal{B}$, with $a_i \subseteq b_i$, $i=1, \dots, m$, noting as ordinary sets, \subseteq replaces \leq , \cap replaces $\&$, \cup replaces \vee , etc.,

$$\text{cond}(g((a_1|b_1), \dots, (a_m|b_m))) = (\alpha|\beta), \quad (3.92)$$

where

$$\alpha \triangleq g(a_1, \dots, a_m), \quad (3.93)$$

$$\beta \triangleq \alpha \cup (g(b_1 \Rightarrow a_1, \dots, b_m \Rightarrow a_m))', \quad (3.94)$$

where here $b_i \Rightarrow a_i = b_i' \cup a_i$.

If g is commutative and associative, then so is $\text{cond}(g): \tilde{\mathcal{B}}^m \rightarrow \tilde{\mathcal{B}}$.

The above development is particularly useful in determining the optimal approximations for naturally extended arithmetic operations, since these, unlike the boolean operations, in general do not possess closure properties over \mathcal{B} . These results lead, in sequence, to the development of random conditional events

and related ideas. (See, in particular, [7], sections 7.3 and 10.4.)

4. PROBABILITY EVALUATIONS AND APPLICATIONS

Following the brief overview of the role of data fusion in developing a generic model for typical C^3 processes in section 1 and the motivations and mathematical and computational structures for conditional events, this section presents the fundamental links for probability evaluation and application to combining of evidence and data fusion.

First note the following theorem:

Theorem 4.1. ([7], Theorems 1.5, 7.1, Remark 1.3)

Let $p: R \rightarrow [0,1]$ be a given probability measure. Then:

(i) With slightly additional conditions placed upon p (non-atomicity), the only possible extension of p to first $p: \tilde{R} \rightarrow [0,1]$, for each $b \in R$, and then to $p: \tilde{R} \rightarrow [0,1]$, where $p((a|b))$ is some fixed function (not dependent upon any given $a, b \in R$) of $p(a \& b)$ and $p(b)$ such that $p((\cdot|b)): \tilde{R} \rightarrow [0,1]$ is a probability measure is

$$p((a|b)) = p(a|b) = p(a \& b)/p(b), \quad (4.1)$$

i.e., ordinary conditional probability must be assigned to conditional events.

(ii) The extension $p: \tilde{R} \rightarrow [0,1]$ is monotone increasing, i.e., for any $a, b, c, d \in R$,

$$(a|b) \leq (c|d) \text{ implies } p(a|b) \leq p(c|d). \quad (4.2)$$

By using the demorgan property among others (see Theorem 3.7(iii) and Corollary 3.1), all computations for probabilities of (compound) boolean functions of conditional events can be reduced to computing probabilities of only conjunctions of conditional events. With this in mind, the following result shows that the measure-free conditional event approach presented here in conjunction with probability evaluations can, in a sense, be identified with a modified form of Approach B given in section 2.3, with the joint probability measure p , in effect, determined through conditional event conjunction and initial probability measure p over R :

Theorem 4.2 ([7], section 10.4, issue (x))

Let $p: R \rightarrow [0,1]$ be a given probability measure, let $b, d \in R$ be arbitrary with $p(b), p(d) > 0$. For each $s, t \in \mathbb{R}$, consider the infinite left rays

$$a_s \triangleq (-\infty, s], \quad c_t \triangleq (-\infty, t]. \quad (4.3)$$

Then, $p((a_s|b) \& (c_t|d))$, as a function of (s, t) , is a legitimate cumulative probability distribution function over \mathbb{R}^2 with $p((a_s|b) \& (c_t|d))$, as a function of s , and $p((c_t|d) \& (b|b))$, as a function of t , being marginal cumulative probability distribution functions over \mathbb{R} .

BASIC PRINCIPLES FOR EVALUATING PROBABILITIES OF CONDITIONED INFORMATION

- (1). Determine whether or not the evidence has truly differing antecedents.
- (2). If the antecedents of the information are identical then apply the usual calculus of relations for PL. For example, suppose the same source, sensor or human, on two different occasions produces an estimate of target location and it is desired to obtain the probability of the disjunction, since if the resulting probability is sufficiently low, no further investigation will be

carried out. If the source error, though possibly independent, is relatively small, then both target estimates can be considered unconditioned information. Or, perhaps, by sheer coincidence, the same error causing mechanism is present and behaves the same way in producing the two estimates, which themselves can differ greatly in terms of random behavior and probability characteristics. Or, further, where Approach B is valid: the actual joint probabilities of the source random mechanism are known, one can then reduce all computations to that involving only PL, because of the common joint antecedent.

(3). But, in general, as in the simplified examples in section 2.1, the antecedents of arriving conditional information arise from widely varying sources, of which little or nothing is known concerning joint probability distributions, yet relatively much is known concerning the individual (consequent) event probabilities.

(4). Isolate all the relative unconditional events a, b, c, d, e, f, \dots making up the forms of the individual conditional probabilities $p(a|b), p(c|d), p(e|f), \dots$

(5). Temporarily ignoring the probabilities, determine just what compound boolean operation is desired. Perhaps one simply is seeking to obtain, in some way, the "joint" probability of $(a|b), (c|d), (e|f)$, i.e., in actuality, the probability of the conjunction $(a|b) \& (c|d) \& (e|f)$, or any other combination as in the examples previously mentioned. Carry out the measure-free computations, based upon the boolean function desired, in accordance with the calculus of conditional event operations as given in section 3.3. Note that any such operation reduces the collection of, possibly many, conditional events to a single conditional event, say $(\alpha|\beta)$, where now α, β are known boolean functions of the input events a, b, c, d, e, f, \dots

(6). Approximate by a single probability measure p , say, in place of all the differently arising probabilities in step 4, so that, in a consistent sense, $p(a \& b) = p_1(a \& b), p(c \& d) = p_2(c \& d), p(e \& f) = p_3(e \& f), \dots$

Obviously, if the very same event, say b is assigned two distinct probabilities from two sources, say $p_1(b)$ and $p_2(b)$, some consensus must be determined before a final common assignment - by perhaps use of least squares, maximum likelihood, or maximal entropy techniques.

In lieu of step (4), it is possible that the conditional probabilities are all given relative to some common joint probability $p_i = p, i=1,2,\dots$, in which case the computational task simplifies.

(7). Simply use eq.(4.1). An example of a very general formula encompassing the probability evaluation of an arbitrary boolean function of multiple conditional events (consider again the comments at the beginning of Corollary 3.1) is given in (3.32).

(8). A related issue to the actual carrying out of the above steps in considering data fusion problems is that of bounding uncertainty or information. One straightforward result is the following, utilizing the results of section 3.3 and the well known Fréchet bounds for probabilities (see, e.g. [12]):

Theorem 4.3.

$$\begin{aligned} \text{For any } a_j, b_j \in R, j=1, \dots, n, \text{ and probability measure } p: R \rightarrow [0,1], \\ \max(0, \sum_{j=1}^n p(a_j | \& b_j) - (n-1)) \leq p(\&_{j=1}^n (a_j | b_j)) \leq \min(p(a_j | b_j)), \\ \max(p(a_j | b_j)) \leq p(\vee_{j=1}^n (a_j | b_j)) \leq \min(1, \sum_{j=1}^n p(a_j | \& b_j)). \end{aligned} \quad (4.4) \quad (4.5)$$

Making the usual definitions for information uncertainty or entropy, for any $a, b \in R, p(b) > 0$,

$$H_p(a|b) \triangleq -\log(p(a|b)), \quad (4.6)$$

it follows that Theorem 4.3 can be converted immediately to bounds on information uncertainty.

Ideally, what is sought is the conditional event calculus - as developed here - analogue of Hailperin's results [12] concerning the bounding of probability values (and hence, correspondingly, information uncertainty) for arbitrary boolean functions of unconditional events when probability bounds are known for the individual (unconditional) events. However, at this point, one must be content with the rather simple results given in Theorem 4.3, until further results are obtained in this area. These properties also tie-in with CPL and the weakened forms of implication mentioned in section 3.5(ii)(XII). More details of this will be presented in future work.

(9). Generalizing the situation described in steps (6) and (7), one can have events of interest $a_i, b_i, c_i, d_i, \dots, \emptyset_i$ (null event), Ω_i (universal event) all belonging to boolean algebra of events R_i corresponding to common probability measure $p_i: R_i \rightarrow [0,1]$, for $i=1,2$, say. But, by making the usual marginal identifications $a_1 = a_1 \times \Omega_2, b_1 = b_1 \times \Omega_2, \dots, a_2 = \Omega_1 \times a_2, b_2 = \Omega_1 \times b_2, \dots$ and assuming that the joint probability measure p_0 of p_1 and p_2 is known, noting that because of (4.1),

$$p_1(a_1 | b_1) = p_0(a_1 \times \Omega_2 | b_1 \times \Omega_2), \dots, \quad (4.7)$$

$$p_2(a_2 | b_2) = p_0(\Omega_1 \times a_2 | \Omega_1 \times b_2), \dots, \quad (4.8)$$

it follows that all of the previous steps are now valid for the situation here with p_0 replacing p , and the marginal identifications for the events.

(10). Higher order conditioning, i.e., when the events in the above steps such as a, b, c, \dots are in actuality in conditioned form already (which may well be a common situation), can be treated in a straightforward way by use of the results in section 3.5(iii).

(11). Finally, it should be remarked that for non-stochastic information, such as that containing linguistic-based evidence, as considered, e.g., in [11], an analogous calculus of conditional forms can be developed, based upon (3.11) and the natural extensions of operations [13]. The corresponding full ALDP's should prove of use in treating combination of evidence problems and data fusion in general.

5. ACKNOWLEDGMENTS

This work was supported jointly by the Naval Ocean Systems Center Program for Independent Research (IR) and the Joint Directors Of Laboratories, Technical Panel for C³, Basic Research Group (JDL, TPC³, BRG).

6. REFERENCES

- Goodman, I.R., "A general theory for the fusion of data", *Proc. 1st Tri-Serv. Data Fus. Symp.*, 1987, 254-270.
- Goodman, I.R., "A probabilistic/possibilistic approach to modeling C³ systems: Part II", *Proc. 1st Symp. C³ Research*, Dec., 1987, 41-48.
- Goodman, I.R., "A probabilistic/possibilistic approach to modeling C³ systems", *Proc. 9th MIT/ONR Workshop, C³ Sys.*, Dec., 1986, 53-58.
- Goodman, I.R. & Nguyen, H.T., *Uncertainty Models for Knowledge-Based Systems*, North-Holland, Amst., 1985.
- Goodman, I.R., "Toward a general theory of C³ processes", *Proc. 2nd Symp. C³ Research*, to be publ., 1988.
- Goodman, I.R., "A measure-free approach to conditioning", *Proc. 3rd AAAI Workshop, Uncert. in AI*, 1987, 270-277.
- Goodman, I.R. & Nguyen, H.T., *A Theory of Measure-Free Conditioning*, submitted for publ. (monograph), 1988-1989.
- Calabrese, P., "An algebraic synthesis of the foundations of logic and probability", *InfSci*, 42, 197, 187-237.
- Schay, G., "An algebra of conditional events", *MAA24(2)*, 68, 334-344.
- Copeland, A.H., "Implicative boolean algebras", *Math. Z.*, 50, 285-290.
- Goodman, I.R., *FACT*, NOSC Tech Doc. 878, March, 1986.
- Hailperin, T., "Probability Logic", *Notre Dame J. Formal Logic*, 25(3), 1984, 198-212.
- Goodman, I.R., "A unified approach to the modeling of uncertainties", to be submitted, 1988.

THIS PAGE LEFT BLANK INTENTIONALLY

A TECHNIQUE FOR AUTOMATICALLY CORRELATING ESM DATA AND RADAR TRACKS

Steven Bennett and George Rebovich, Jr.

**The MITRE Corporation
Bedford, Massachusetts 01730**

ABSTRACT

Electronic support measures (ESM) systems are passive surveillance sensors for determining emitter and platform information from a target's radio frequency emissions. When paired with a radar track on a target, this information can be valuable in identifying it. The pairing could be done manually by a surveillance operator or automatically by an operational computer program.

A technique for automatically correlating ESM reports and radar tracks has been developed which shows the potential for good to excellent performance while requiring modest data processing capabilities.

The approach is to employ a recursive, multiple hypothesis track-to-ESM report correlation testing technique which uses probability thresholds to prune unlikely hypotheses over a number of association attempts. The distinguishing features of the technique are its adaptive azimuth windows which are used in data association and the decision confidence logics which are used to winnow unlikely candidate hypotheses.

Algorithms for this technique have been developed and their performance evaluated by simulation. The paper outlines the salient features of the correlation process and presents the results of the simulation.

A TECHNIQUE FOR AUTOMATICALLY CORRELATING ESM DATA AND RADAR TRACKS

Steven Bennett and George Rebovich, Jr.

The MITRE Corporation
Bedford, Massachusetts 01730

INTRODUCTION

Increases in the electronic countermeasures (ECM) threat to active radar and identification friend or foe (IFF), together with advances in electronic support measures (ESM) technology are making passive sensors an attractive adjunct to active sensors for surveillance and identification in tactical air defense systems. Integrating a new sensor, such as ESM, into an existing tactical command and control (C^2) system can present a number of special challenges when addressing the requirement for fusing data from the newly incorporated sensor and an existing one.

The solution to the fusion problem for this situation must normally satisfy a highly constrained set of conditions. It must contend with design choices, such as selection of tracking coordinate system, filters and state variables, which were made without consideration for a multisensor data fusion requirement. Frequently, the solution must be capable of being implemented within existing computer and operational computer program architectures and resources. Above all, the solution must provide tactically significant information in time for the user to take advantage of it.

This paper outlines an automated fusion technique for correlating ESM data and radar tracks. The technique has the potential for achieving good to excellent performance while requiring modest data processing capabilities. In principle, the technique can be implemented in a processor which supports an automated, real-time radar tracking function.

The technique is applicable to both airborne and ground C^2 systems which have a requirement for correlating ESM data and radar tracks. The more general airborne application is presented here; the ground-based application can be considered as a special case in which there is no ownship motion.

The remainder of the paper discusses the problem and describes the salient features of the fusion technique. The results of a simulation are used to present an example of the technique and address potential performance.

PROBLEM

The particular problem addressed here is that of automatically correlating radar tracks with ESM system reports. Both sensors are taken to be integrated into an airborne C^2 platform. The

radar tracker considered is a conventional track-while-scan (TWS) system which uses Kalman filtering to estimate target position and velocity in a Cartesian coordinate plane. The ESM system is a general purpose, wide-band passive surveillance sensor for determining angle of arrival (AOA) to a target as well as emitter and platform information from the target's radio frequency (RF) emissions. If an ESM report could be paired with a radar track on a target, the emitter and platform information can be valuable in identifying it. For example, the ESM report could indicate the presence of a hostile fighter.

The hypothetical air picture in figure 1 indicates the nature of the correlation problem. The airborne C^2 system is at the center of the display. Radar returns which have been correlated with tracks surround it throughout the surveillance volume (all tracking symbology has been shut off). Reports on ESM observations are

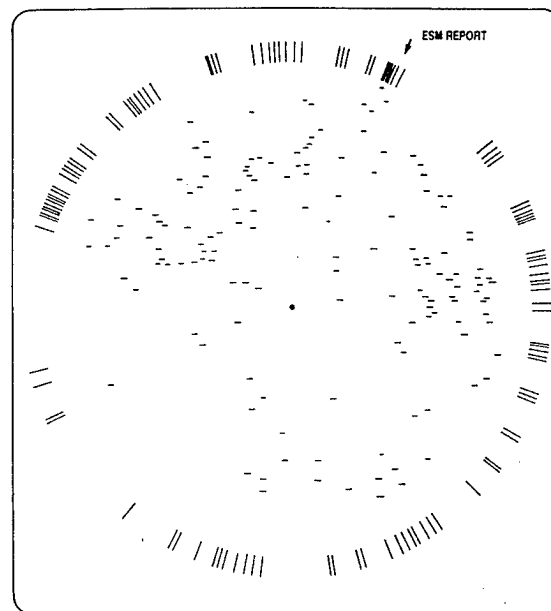


Figure 1. Hypothetical Radar and ESM Air Picture

displayed as short strobes on the periphery of the display. Since angle information is the primary discriminant for correlating ESM reports and radar tracks, the correlation cannot always be made unambiguously or with high confidence in one attempt.

Note that the problem addressed here is correlating the correct radar track with an ESM report of interest. The technique discussed below can be applied to the converse problem, *mutatis mutandis*.

CORRELATION TECHNIQUE

The approach is to use a recursive, multiple hypothesis track-to-ESM report correlation testing technique which employs probability thresholds to prune unlikely hypotheses over a number of association attempts.

A distinction is made in this paper between association and correlation. Association is a test to determine whether a radar track and ESM report fall in an azimuth window. Correlation is the recommended assignment of a radar track to an ESM report. Correlation is, therefore, a decision which is made on the basis of a number of association attempts.

The azimuth windows used in each association attempt are sized adaptively using ESM system AOA measurement error statistics and the real-time estimates of radar tracker performance provided by the Kalman filter covariance matrix. This careful sizing of the association windows allows the accurate prediction of the association performance between the correct track and the ESM report over a number of association attempts. In addition to winnowing unlikely hypotheses, the association performance thresholds can be used to attribute a level of confidence to any surviving hypotheses.

The major functional elements of the correlation technique are initiation, gating, hypothesis selection, testing, deletion and hypothesis declaration. These elements are described below.

Initiation

Initiation starts the correlation process by specifying ESM information of interest and a nominal AOA. This can be done in many ways. One way would be for the surveillance operator to mark a particular ESM report for processing by console switch actions. Alternatively, correlation processing could be automatically initiated on ESM reports passing filters for geographical sectors and emitter type.

Gating

The purpose of gating is to eliminate the vast majority of highly unlikely correlation hypotheses. After the correlation process is initiated, the radar tracks are filtered using a constant width azimuth window centered about the initiation AOA. Tracks falling outside the window represent improbable hypotheses in that they are unlikely to have generated the ESM report and are dropped from further processing. An azimuth window width of from 5 to 10 times the standard deviation of the combined nominal ESM AOA measurement error and angular component of the radar track estimation error has produced consistently good gating results.

Hypothesis Selection

The remaining tracks represent tentative correlation hypotheses which are subjected to a more rigorous azimuth association test. Tracks passing this test are promoted to candidate hypothesis status. This test uses adaptive azimuth windows which account for the angular component of the radar track position error and ESM report AOA error.

For each track hypothesis making it to this point in the processing, the positions of the airborne C² platform and the radar track are linearly predicted and the 1-sigma error ellipse defined by the position elements of the radar track's covariance matrix are propagated to the time of the ESM report. The predicted positions are used to calculate a predicted track azimuth which is then used to determine the azimuthal component of the propagated error ellipse. This part of the processing is depicted in figure 2.

In the particular example shown in figure 2, the current C² platform position information (from its on-board navigation computer) is the most recent. ESM report data (from the ESM subsystem active emitter file) are next and the smoothed track data (from the radar track table) are the most stale.

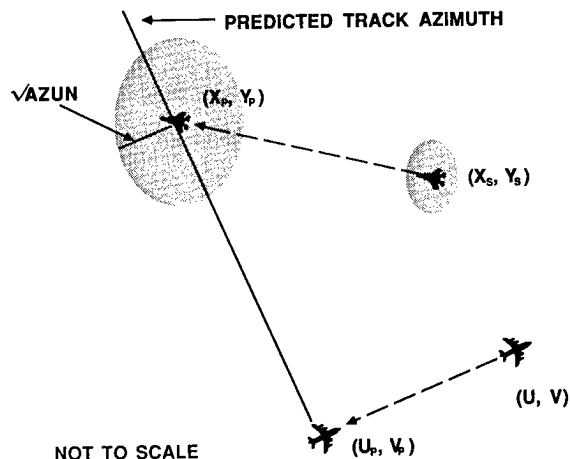


Figure 2. Azimuth Uncertainty of Predicted Track Position

As depicted, the current C² platform position coordinates (u,v) are predicted back in time to (u_p, v_p) using the navigation system estimates of ownship velocity. The position coordinates of the track's smoothed state vector (x_s, y_s) are predicted forward in time to (x_p, y_p) using the smoothed velocity components. That is:

$$u_p = u + \dot{u}\Delta t_1 \quad v_p = v + \dot{v}\Delta t_1$$

$$x_p = x_s + \dot{x}_s\Delta t_2 \quad y_p = y_s + \dot{y}_s\Delta t_2$$

where

$$\Delta t_1 = t_{\text{ESM}} - t_{\text{NAV}} \text{ and } \Delta t_2 = t_{\text{ESM}} - t_s.$$

The predicted track azimuth is calculated as:

$$\text{Arctangent } (x_p - u_p)/(y_p - v_p).$$

The 1-sigma position error ellipse defined by the track's covariance matrix is linearly propagated to the ESM report time according to:

$$P' = \Phi P \Phi^T$$

where P and P' are the smoothed and predicted covariances and Φ is the transition matrix. The ellipses are also depicted in figure 2.

The azimuthal component of the propagated 1-sigma error ellipse is then calculated from the predicted azimuth and the propagated 1-sigma error ellipse as:

$$\frac{(y_p - v_p)^2 P'[1,1] - 2(y_p - v_p)(x_p - u_p)P'[1,2] + (x_p - u_p)^2 P'[2,2]}{[(x_p - u_p)^2 + (y_p - v_p)^2]^2}$$

where

$$P'[1,1] = P[1,1] + P[3,3](\Delta t)^2 + 2P[1,3]\Delta t,$$

$$P'[1,2] = P[1,2] + \Delta t(P[2,3] + P[1,4] + \Delta tP[3,4])$$

and

$$P'[2,2] = P[2,2] + P[4,4](\Delta t)^2 + 2P[2,4]\Delta t.$$

This is AZUN which represents the variance of the predicted track azimuth error. The standard deviation, \sqrt{AZUN} , is depicted in figure 2. The notation, $P[\text{row}, \text{column}]$, is used to indicate the elements of the covariance matrix and adheres to the usual conventions. That is, $P[i,j]$, for $i = 1, \dots, 4$, is the variance of the state variables x, y, \dot{x}, \dot{y} , respectively.

The variance of the track azimuth error is then added to the variance of the ESM report azimuth error, MEASUN, to calculate the variance of the combined track azimuth and ESM report azimuth errors. This combined variance is used to size an azimuth association window, CORWIN, according to:

$$\text{CORWIN} = 2 \times \sqrt{(\text{AZUN} + \text{MEASUN})}.$$

The association window is then centered about the ESM report AOA as depicted in figure 3. If the predicted track position falls in the association window, the track is tentatively paired with the ESM report and an association counter is initialized for that track. Note that under the assumption that the azimuth errors in the radar track and ESM report are independent and normal with zero mean, CORWIN represents a 2-standard deviation window for a zero mean, normal probability distribution. As a result, the correct track can be expected to fall in the association window about 95 percent of the time.

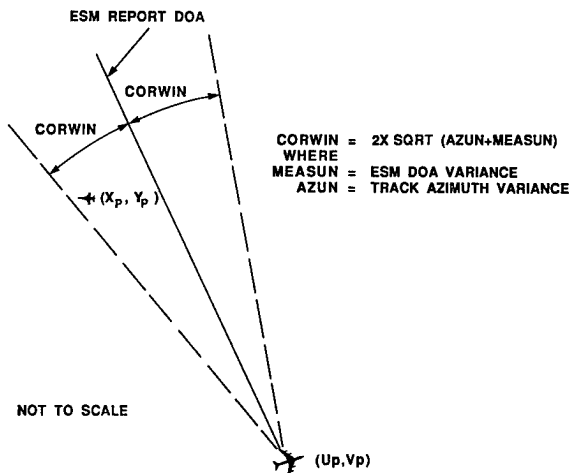


Figure 3. Azimuth Correlation Test

MEASUN is the variance of the ESM report AOA error. ESM AOA measurement performance is normally amenable to table look-up techniques for the purpose of building correlation windows. Specific values for the table would be developed from ESM system specification requirements or test data.

Tracks which associate with the ESM report at least once in the first three attempts are promoted to candidate hypothesis status. Using this criterion, the only way the correct track would not be promoted is if it failed to pass all three association attempts. Under the assumption that the association attempts are independent events,

then the probability of missing all three is $(1.0 - .95)^3 = 1.25 \times 10^{-4}$. The probability of the complementary event, promoting the correct track, is therefore $1.0 - 1.25 \times 10^{-4} \approx 0.99$.

Hypothesis Testing and Deletion

After they are promoted, candidate track hypotheses are processed against the ESM report over time to generate association/missed-association histories. This is done by using the technique just described in the hypothesis selection section.

The association history of a hypothesis begins with process initiation. The history ends after a predefined maximum time or when a hypothesis is eliminated as being unlikely. After each association cycle, the association histories of the hypotheses are updated to reflect the most recent outcome. The probabilities of the updated hypotheses are recomputed and used to evaluate whether a hypothesis is eliminated or retained. This is the recursive feature of the correlation technique. Two criteria are used to eliminate candidate hypotheses: exceeding a cumulative missed association threshold or three consecutive missed associations.

The cumulative miss threshold represents a level of decision confidence for retaining or eliminating candidate hypotheses over a sequence of association attempts. The threshold is determined by estimating the number of times the correct track can be expected to fall in the association window of an ESM report over a sequence of association attempts. Under the assumption that individual association attempts are independent events, the expected number of associations over multiple attempts is a binomial random variable. The expected number of associations can be calculated from the expected performance of a single attempt and the probability function of the binomial distribution according to:

$$b(x; n, p) = \binom{n}{x} p^x (1-p)^{n-x}$$

where x is the number of successful associations, n is the number of association attempts and p is the probability of correct association for a single attempt. Strictly speaking, the assumption of independence between successive association attempts is not valid because of the error correlation between radar track estimates from smoothing to smoothing. In practice, however, this assumption has negligible impact on correlation performance because of the relative magnitude of the ESM report and radar track errors.

To see how the cumulative miss threshold is computed and used, consider the previously defined 2-sigma association windows. Recall that, with this window, there is a 95% probability of associating the correct track with the ESM report in a single attempt. Using the binomial probability function and noting that the event "at least 13 associations in 15 attempts" is equivalent to the logical union of the three mutually exclusive events "exactly x associations in 15 attempts" for $x = 13, 14$, and 15, there is 97% probability that the correct radar track will associate with the ESM report at least 13 times in 15 attempts (two or fewer misses). A track hypothesis processed at the 97% confidence level, therefore, would be eliminated if the third missed association occurred at the fifteenth association attempt and if it had not been eliminated earlier. Hypotheses processed at higher levels of confidence (e.g., 99%) would require more misses before being eliminated; hypotheses processed at lower levels (e.g., 90%) would be eliminated with fewer misses.

An attractive feature of this approach is that the missed association thresholds corresponding to a confidence level can be pre-computed and employed in table look-up techniques rather than computed in real-time. An example for the 95% confidence level is shown in Table 1. A hypothesis would be evaluated after each association attempt and eliminated if it exceeded the permissible number of missed associations.

Table 1. Missed Association Thresholds for a 95% Level of Confidence

Association Attempt	Permissible Missed Associations
1	0
2	1
3	1
4	1
5	1
6	1
7	1
8	2
9	2
10	2
11	2
12	2
13	2
14	2
15	2
16	2
17	3
18	3
19	3
20	3

The three consecutive miss criterion is an ad hoc technique which has been employed to handle situations in which the missed association history of an incorrect hypothesis fell far below threshold, making it unlikely to be eliminated quickly. This can occur when the airborne C² platform, the correct track and an incorrect track are colinear for some time. The three consecutive miss criterion could eliminate an incorrect hypothesis quickly while posing a small risk of eliminating the correct hypothesis.

Declaration

Processing candidate hypotheses continues until one of the following conditions is satisfied:

1. After a predefined minimum number of processing cycles, all but one candidate hypothesis is eliminated, or
2. Processing has exceeded a predefined maximum time or number of processing cycles, or
3. All candidate hypotheses are eliminated.

If the first condition is satisfied, the hypothesis is accepted and the surveillance operator is notified of the recommended track-to-ESM report correlation. If the second condition holds, there are multiple hypotheses remaining and the operator is notified of the ambiguous correlations. If the third condition holds, the operator is notified that no correlations were made.

AN EXAMPLE

Returning to the hypothetical air picture depicted in figure 1, assume the ESM report marked by the arrow unambiguously indicates a hostile fighter. There are a number of radar tracks which could reasonably be paired with the ESM report.

After correlation processing was initiated on this ESM report, all but eight tracks were eliminated by gating. These eight tracks are depicted notionally in figure 4 along with their velocity vectors. Association attempts were made every five seconds, corresponding

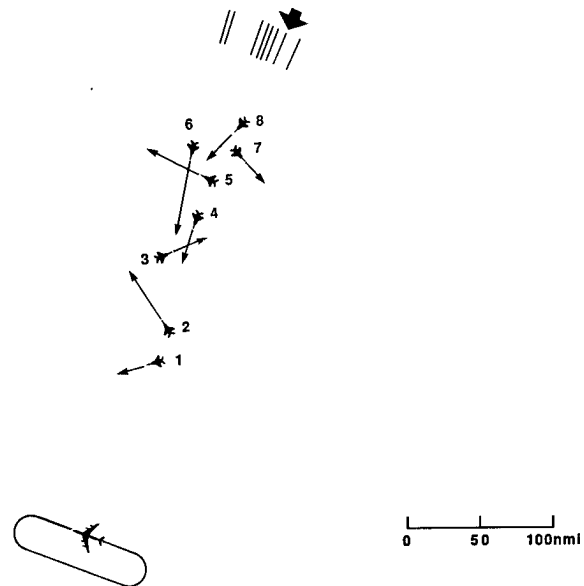


Figure 4. Hypothetical Correlation Problem

to the refresh rate which was posited for the ESM system. Tracks 1, 2 (the correct track) and 7 each associated at least one time in the first three attempts and so became candidate hypotheses. Testing these hypotheses at the 95 percent confidence level, track 1 was eliminated 35 seconds after process initiation for exceeding the cumulative miss threshold and track 7 was eliminated 10 seconds later for exceeding the cumulative miss threshold and three consecutive misses simultaneously. Track 2 was therefore uniquely and correctly paired with the ESM report in 45 seconds.

PERFORMANCE TRENDS

The automated process was simulated for each of the eight ESM reports in figure 4 over a range of ESM AOA accuracies and confidence levels for eliminating candidate hypotheses.

Certain ground rule assumptions applied to the simulation. Targets did not maneuver during the scenario. The radar antenna was assumed to rotate in azimuth mechanically with a 10 second period. Radar detection performance was assumed to be adequate to support tracking using conventional Kalman filtering techniques. The tracker predict-correlate-smooth cycle was assumed to be 10 seconds corresponding to the radar antenna rotation period.

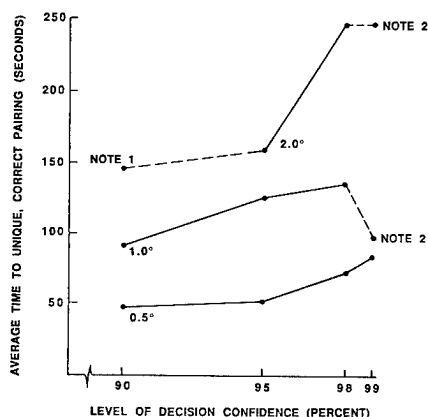
The tracking filter was taken to be implemented as two independent channels — one for the x-variables and one for the y-variables. This is the approach frequently taken in real-time applications where any loss in track estimation accuracy is compensated by a significant reduction in processing required to execute the filtering. Note that, in this approach, the cross-correlation elements of the fully coupled covariance matrix are not computed and so are unavailable for determining AZUN. No compensation was made for this in sizing the association windows. As a result, the cross-correlation elements were set to zero when computing AZUN in the simulation.

Regarding the ESM system, each target was assumed to emit a signal throughout the scenario that was detected by the ESM system and was distinguishable from the other target signals. It was assumed that the ESM system could be directed to search the RF spectrum of interest so that emitter reports in the active emitter file would be updated every 5 seconds. Actual update times were modeled as occurring randomly in each 5 second update cycle to

account for changes in the ambient emitter activity, the sequence and timing of the ESM system receiver search strategy and signal processing activities.

The radar and ESM system were assumed to operate independently. As a result, there was no time synchronization between the update of the radar track file and the ESM system active emitter file.

Correlation performance is summarized in figure 5. There is a performance curve for each level of ESM AOA accuracy considered. In the majority of scenarios, seven of eight radar tracks were uniquely correlated with the correct ESM report and so average time to correct correlation is used to indicate the relative performance among the AOA accuracies and confidence levels considered. The data for scenarios in which seven of eight unique, correct correlations were achieved are joined by solid lines. The one scenario in which the correct track was eliminated and the two scenarios in which six of eight unique, correct correlations were achieved are included for completeness. These results are joined to the other data by broken lines. Correlation performance was degraded in these scenarios and so average time to correct correlation is not an adequate indicator of relative performance.



NOTE 1 - ONE CORRECT TRACK ELIMINATED IN THIS SCENARIO.
NOTE 2 - SIX OF EIGHT CORRECT, UNIQUE PAIRINGS; 2 CORRECT, AMBIGUOUS PAIRINGS

Figure 5. Correlation Performance Summary

Several trends are indicated. Good to excellent correlation results appear achievable over the ranges of confidences and AOA accuracies considered.

The correlation results were the same for the four cases in which ESM AOA accuracy was 0.5 degrees (1-sigma). There were seven of eight unique, correct correlations. However, as the confidence required to make these correlations increased, so did the time to achieve them. The average time ranged from 48 seconds at the 90% confidence level to 85 seconds for the 99% confidence level.

The final correlation results for the 1.0 degree AOA accuracy scenarios were the same as those for the 0.5 degree scenarios at the 90, 95, 98 percent levels of confidence. There were seven of eight unique, correct correlations. However, the average time to achieve these results increased. For example, at the 95% confidence level it took 125 seconds compared to 52 seconds for the 0.5 degree scenario. More time was required for the targets to separate enough to allow the incorrect tracks to be confidently disassociated from the 1.0 degree accuracy ESM reports. At the 99% confidence level, the hypothesis which paired track 2 with ESM report 4 could not be eliminated. Therefore, six of eight tracks were uniquely and correctly correlated with their ESM reports while two were correctly but ambiguously correlated.

Requiring very high confidence in the results increased the number of ambiguous correlations in this case.

At the 95% level of confidence, the final correlation results for the 2.0 degree accuracy scenarios were the same as those for the 0.5 and 1.0 degree scenarios — seven of eight unique, correct correlations. The average time to achieve the results was 159 seconds compared to 125 seconds for the 1.0 degree scenario and 52 seconds for the 0.5 degree scenario.

The correlation results did not differ at the 98% level of confidence although the average time to achieve them increased to more than four minutes (247 seconds). At the 99% level, the hypothesis which paired track 2 with ESM report 4 could not be eliminated. This decreased the unique correlations from seven to six and increased the ambiguous ones from one to two.

The 2.0 degree scenario demonstrated that processing at lower levels of confidence increases the risk of eliminating the correct track. At the 90% level, the correct hypothesis which paired track 4 with ESM report 4 was eliminated for exceeding the cumulative miss threshold. At that point, the only hypothesis regarding ESM report 4 which remained was the one pairing track 2 with it, although this hypothesis would have been eliminated 15 seconds later, also for exceeding threshold.

In general, the better the ESM AOA accuracy, the better the correlation performance, but acceptable performance seems likely up to 2.0 degrees AOA accuracy. Note that these results were achieved without compensating for the unavailable cross-correlation elements of a fully coupled Kalman filter or the invalid assumption of independence between association attempts. In general, these would have to be considered. In this application, however, they were of negligible impact because the ESM AOA errors generally dominated the radar track AOA errors.

Selecting a confidence level for processing involves striking a balance between tolerating some ambiguous or incorrect correlations and timeliness of results. This balance may ultimately be decided by operational requirements in specific situations. However, the 95% level of confidence seemed to be the best compromise in the examples considered in this paper.

SUMMARY

This paper has outlined an automated technique for correlating ESM data and radar tracks which has the potential for achieving good to excellent performance while requiring modest data processing capabilities. It is a suitable baseline for C² systems where the fusion capability must be implemented with nominal computer resources.

The technique is a recursive, multiple hypothesis track-to-ESM report correlation algorithm which employs probability thresholds to prune unlikely hypotheses over a number of association attempts. The association windows are carefully sized to eliminate unlikely hypotheses and attribute a level of confidence to surviving hypotheses.

A distinctive feature of the technique is the capability to set the level of confidence at which processing is performed. Setting of the confidence level could be implemented as an adaptation parameter which would be set to a default value or selected by the operator before or during a mission. This permits the human to make the important operational decision which strikes the balance between timeliness of results and tolerating some ambiguous or incorrect correlations while leaving the execution of the correlation function to an operational computer program.

MULTI-SENSOR TARGETING AND TRACKING

Dr. Charles P. Bernardin and Dr. Yun-Kung Jack Lin

Texas Instruments
Dallas, Texas

ABSTRACT

Effective weapon and surveillance systems require accurate spatial location information for tactical functions such as fire control and early warning. Jamming, adverse weather, and deceptive target maneuvers often found in the air/land combat environment present a challenge in determining a target's position in space. Multi-sensor fusion can improve the reliability of the tracking system; because of their multi-spectral nature, such systems are more immune to unfavorable environmental conditions. The fusion of 2-D passive sensors with 3-D active sensors can provide: 1) more covert surveillance, 2) better target identification, 3) improved target track-file association, 4) higher target detection probability with fewer false alarms, and 5) more precise estimates of the 3-D target position. This study focused primarily on the last advantage of sensor fusion, i.e., 5). Sensor integration is considered in terms of fusing two- and three-dimensional spatial measurements. The resulting improvement in estimating 3-D position provides a quantitative metric of fusion performance. Significant performance improvement in tracking accuracy achieved in a simulation for both helicopter and fixed-wing targets is reported in this paper.

MULTI-SENSOR TARGETING AND TRACKING

Dr. Charles P. Bernardin and Dr. Yun-Kung Jack Lin

Texas Instruments
Dallas, Texas

INTRODUCTION

Emerging military systems require optimal use of multiple sensors and information sources in order to meet mission objectives. Their survivability must be enhanced through longer standoff, significantly reduced false alarms, improved target detection and identification, and operator workload reduction. These are all key to reducing mission timeline and consequent platform exposure (Ref.1). Algorithms and processing techniques that accomplish the synergistic combining of sensor inputs across the frequency spectrum (acoustic, IR, TV, radar), as well as stored data inputs such as digital map, have not yet reached operational maturity.

This paper presents an algorithm development and evaluation effort for a multi-sensor fusion simulation system. This VAX-based system is easily adapted to handle a wide spectrum of tactical applications such as those found in ground-to-air and air-to-air scenarios. Simulation results presented here focus on a multi-sensor three-dimensional tracking solution in an air defense scenario.

SENSOR FUSION SIMULATION SYSTEM AND ALGORITHM DESCRIPTION

The sensor fusion simulation system shown in Figure 1 has five major functions: report generation; association; false-alarm reduction; ID improvement; and tracking. We investigated candidate approaches for the last four functions and selected algorithms for implementation based on performance and processing complexity.

Sensor Report Generation

We selected a sensor suite that would be applicable to an integrated air defense system. This sensor suite was composed of radar, infrared search and track (IRST), radio frequency direction finder (RFDF), and a nonimaging (NI) sensor. Mission profiles for three target types in a simulated scenario were established using the Texas Instruments developed Composite Air-Surface Engagement Simulation (CASES), a computer engagement model that was designed to evaluate the effectiveness of surface-based air defenses,

air vehicle survivability, and air-to-surface weapon and sensor requirements. Sensor reports for each of six sensor sites were then generated based on target and sensor performance parameters.

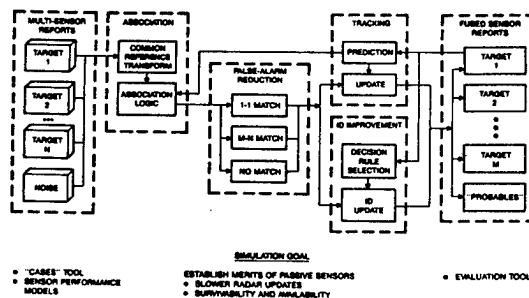


Figure 1. Sensor Fusion Simulation System

Association

The association function (Ref. 9) is a front-end process that matches each of the incoming sensor reports with the fused tracks already in the track file database. Its decision outputs declare one-to-one, multiple, or no match. Ambiguities from multiple or no matches will be resolved in the false-alarm reduction function.

The procedure calls for a common reference transformation (CRT) in both space and time and a hypothesis generation/evaluation/selection process. The CRT allows intersystem cueing, noncollocated sensor deployment, and integration flexibility of nondevelopmental item (NDI) elements. A hypothesis set is generated by a spatial distance thresholding criterion. ID and kinematics from sensors, along with the a priori sensor performance, define a rule-based expert that evaluates and selects association hypotheses.

False-Alarm Reduction

Four false-alarm reduction algorithms were considered for the fusion system: Shafer-Demp-

ster (Ref. 6), joint statistical detection (e.g., Bayesian), Pd/Pf ratio (Pd = probability of a detection, Pf = probability of a false alarm), and M-or-more detection. Based on performance and processing capabilities, the last approach was chosen since it provides a simple solution when noise is uncorrelated across sensors. This was a reasonable assumption for the sensor-suite selected in this study.

Multi-sensor detection performance can be modeled as N repetitions of Bernoulli trials (where N is the number of sensors). If all N trials (sensors) are independent and the probabilities (detection and false alarm) are the same (i.e., N identical but distinct sensors), the detection performance of the N sensors can be described by the binomial distribution. If the probabilities are not the same but are constant (i.e., N different sensors), the detection performance of the N sensors can be described by a generalization of the binomial distribution. In either case, the question that seems most appropriate is "Did M or more sensors detect?". If they did, a detection is reported; otherwise, there is no detection.

Our results showed that the best choice for M is approximately N/2; i.e., if half or more sensors detect, then the fused decision is a detection. This fused decision will always have improved detection and false-alarm performance (i.e., increased Pd/Pf ratio) compared with any of the individual sensors.

Identification Improvement

ID improvement occurs at three levels. The first level is an automatic fusion process of ID data from the NI sensor reporting rotary-wing targets and the RFDF identifying radiating targets; both sensors are capable of identifying multiple targets. At the next level, some high-priority targets that need further ID clarification are processed by other dedicated ID sensors, such as FLIR and other nonimaging ID sensors, that may be included in the sensor suite; the dedicated sensors must be cued for each target. The third decision is made at the operator level: the operator evaluates the machine recommendations, with aids from other intelligence data about the target, to derive the final decision.

Several ID decision techniques were investigated as an aid to the operator for the first-level and third-level processes. The techniques include Shafer-Dempster rules of combination (Ref. 6), joint statistical classification, and rule-based expert. The latter was chosen because of its capability to handle incomplete sensor reports while making a list of prioritized probable IDs to the operator.

INTEGRATED TRACKING SOLUTION

Track Filter Selection

Several multiple-target, multiple-sensor tracking approaches have been reported in the literature (Refs. 2-8). Nearly optimal tracking performance can be achieved with the Kalman class of filters, depending on the target maneuver model and the specific target maneuver; however, to implement a 3-D optimal Kalman filter, all six target states must be

available. Passive sensors offer no range information, and some tactical situations may not warrant operation of radar, even if it is available. Jamming and target inter-visibility problems may also restrict range measurements. The tracking approach must be flexible enough to incorporate passive bearing-only measurements with range measurements when they are available. For these reasons, a simple alpha-beta filtering approach that provides flexibility in system design and facilitates a direct evaluation of the effects of fusing passive bearing-only sensors with an active radar is adopted here.

Integrated Tracker Approach

The multi-sensor tracking system has several features: 1) angle reports from different sensors that occur simultaneously in time are converted into a single fused angle report; 2) azimuth and elevation reports are tracked independently when no range information is available; 3) when radar measurements are available, the polar information is transformed into the Universal Transverse Mercator (UTM) cartesian reference frame and tracked with an uncoupled 3-D alpha-beta tracker; and 4) cartesian position is transformed back into the polar reference frame and the smoothed bearing estimates are used to update the angle tracker described in 2).

Figure 2 is a block diagram of the multi-sensor tracking system. First, simultaneous bearing reports from sensors are fused. Mathematically, this corresponds to forming a minimum mean square, unbiased estimator, X , of target bearing given simultaneous sensor measurements, Z_i , of different variances (sensitivities) σ_i^2 . Assuming random, independent, unbiased measurement errors, the solution is:

$$X = \sum_{i=1}^n K_i Z_i \quad (1)$$

where

$$K_i = \left[\sigma_i^2 \sum_{j=1}^n \frac{1}{\sigma_j^2} \right]^{-1} \quad (2)$$

The error of the estimate is:

$$\text{MSE} = \left[\sum_{j=1}^n \frac{1}{\sigma_j^2} \right]^{-1} \quad (3)$$

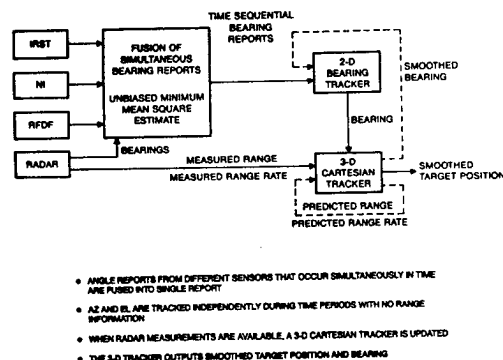


Figure 2. Block Diagram of Multi-Sensor Tracking System

The next block represents a bearing-only tracker. Here, the time-sequential bearing estimates are fused using a 1-D alpha-beta track filter. The weights (alphas and betas) are chosen based on the measurement errors (i.e., the σ_1^2). The output of this tracker is a smoothed estimate of the target's bearing (azimuth and elevation).

When radar measurements are available, 3-D target position is updated. Measured range and range-rate are combined with the improved bearing estimates from the bearing-only tracker with a polar-to-cartesian transform. The target's position in UTM coordinates is estimated and tracked. The cartesian state vector is then transformed back to a polar vector, and the 3-D bearing values are input into the bearing-only tracker to reduce angle drift.

MULTI-SENSOR TRACKING SIMULATION

Sensors and Targets Assumptions

The simulation sensor suite consists of a radar along with three passive sensors: IRST, NI, and RFDF. To simplify the simulation, the timelines of all of the sensors are assumed to be synchronized, although each with its own update rate (4.0, 1.5, 1.0, and 1.0 seconds, respectively). We assume that the sensors are collocated, the targets are not severely maneuvering, the jamming environment is benign, and the Hind-D gunship has a multi-mode fire control radar in its undernose sensor pack.

Since the simulation baseline and 3-D tracking performance is primarily driven by the radar track-while-scan requirements, some of the radar assumptions are discussed here. The antenna is assumed to be a flat-plate 1 meter in elevation by 2 meters in azimuth. The operating frequency is 2 ghz. Monopulse tracking in azimuth and elevation is assumed, yielding effective azimuth and elevation beamwidths of 0.5 degree and 1 degree, respectively. The radar is assumed to be forward-looking (180 degrees in azimuth) and restricted between +45 and -45 degrees in elevation. This gives 450 spatially distinct beams (i.e., potential targets) that can be tracked. The PRF is 7.5 khz based on the maximum range of the radar (20 km). The range resolution is 30 meters and the range rate resolution is 2.5 meters per second.

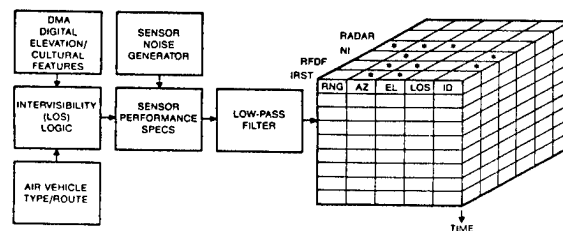
Sensor and Target Models for Reports Database Generation

Table 1 contains the performance characteristics of the sensor suite in terms of sensor update rate, detection range, pointing accuracy, range, and range rate. As depicted in Figure 3, sensor reports were generated in a simulated scenario in which a Flogger, a Frogfoot, and a Hind-D flew the projected profiles in the Defense Mapping Agency's (DMA's) Fulda Gap map area. The Flogger flew 200 meters above ground level (AGL) at 450 knots, while the helo flew 50 meters AGL at 80 knots and occasionally at treetop levels. These profiles were a compromise between the operational requirements and conditions under which a meaningful data set of sensor reports can be derived (the helo flying below 30 meter AGL, for example, did not have line of sight with the sensors for the entire route). Both the elevation and cultural features were used

to determine the target and sensor intervisibility. Sensor inaccuracies were modeled with lowpass-filtered white noise. In this way, a data set of sensor reports for the Hind-D and the Flogger for time periods of 1133 and 101 seconds, respectively, was generated for the multi-sensor tracking simulation evaluation.

Table 1. Sensor Performance Model
(1-SIGMA for pointing accuracy)

Sensor	Update (sec)	Detection Range (km)	Bearing (degrees)		Range (m)	Range Rate (m/sec)
			Az	El		
Radar	4.0	20	0.43	0.86	30	2.5
IRST	1.5	15	0.12	0.12	--	---
RFDF	1.0	20	0.5	0.5	--	---
NI	1.0	15 (incoming, 8 outgoing)	5.0	---	--	---



- MULTI-SENSOR, MULTI-TARGET, AND MULTI-PLATFORM DATA ARE ORGANIZED IN 3-DIMENSIONAL ARRAYS
- THESE SYNTHETIC DATA BASES ALLOW SIMULATIONS OF INTER-PLATFORM CONFIGURATIONS FOR APPLICATIONS AS FAADC²
- ASSUMPTIONS
 - BENIGN JAMMING
 - RADAR-EQUIPPED HIND-D GUNSHIP
 - COLLOCATED SENSORS (WILL EXTEND TO NONCOLLOCATED CASES)

Figure 3. Sensor Report Generation

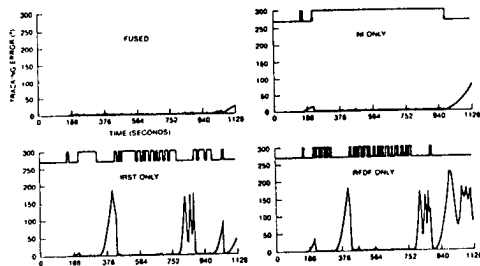
It should be noted that we attempted to get sensor reports as close as possible to operational realities. As a result, the RFDF reports reflected that the target had operated its radar between the track-while-scan and the single-target-tracking modes, as shown in Figure 4. The top of each subfigure depicts the sensor report indicators. The indicator levels stayed high for received reports.

SIMULATION RESULTS

Three Dimensional Tracking Results

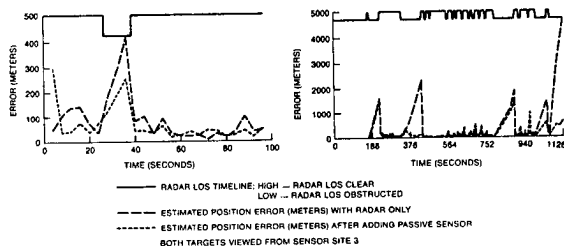
Simulation results for tracking a fixed wing target are shown in Figure 5(A): the topmost curve represents the timeline for the radar (high signifies that the radar LOS is unobstructed; low signifies that the radar LOS is obstructed); the solid curve below is a plot of the error in estimating fixed-wing position with just the radar; the dashed curve is a plot of the error that results from using both the radar and the passive sensors. Note

that sensor fusion tends to improve estimates of target location, especially when the radar LOS is obstructed.



- SIGNIFICANT IMPROVEMENT IN ESTIMATING BEARING, ON AVERAGE: 6.5° (FUSED) VERSUS 34.8° (IRST ONLY)
- EXTENDED AMOUNT OF TIME DURING WHICH TARGET'S BEARING CAN BE TRACKED FROM 546 TO 832 S
- IMPROVED BEARING PREDICTIONS FOR PERIODS OF NO RADAR REPORTS FACILITATE REACQUIRING TARGET

Figure 4. Fused Bearing Tracking on Helicopter (Sensor Site 3)



- LOS TO FIXED WING IS OBSTRUCTED DURING 22-S TO 40-S INTERVAL
- BEARING FROM PASSIVE SENSORS IMPROVES ESTIMATES OF 3-D POSITION BY ABOUT 40% OVER THIS INTERVAL
- LOS TO HELICOPTER IS OBSTRUCTED DURING 300-S TO 400-S INTERVAL
- BEARING FROM PASSIVE SENSORS IMPROVES ESTIMATES OF 3-D POSITION BY ABOUT 57% OVER ENTIRE FLIGHT PATH

Figure 5. 3-D Tracking for Fixed Wing and Helicopter Targets

Similar results for tracking a helicopter appear in Figure 5(B): the dotted curve is a plot of the error that results from using both the radar and the passive sensors (fusion). Note that sensor fusion generally improves estimates of target location. In some of the cases, the fused performance is actually worse (e.g., $t = 900$ seconds, $t = 960$ seconds); this is due to fused bearing tracker errors during target maneuvers and is mostly a result of the NI sensor, the least accurate sensor in the suite. A number of maneuver-adaptive approaches could be used to solve these specific problems.

The helicopter results are further summarized in Table 2. Two conditions are represented: the radar LOS unobstructed and the radar LOS obstructed. When the radar is off, the tracker continues to estimate target position with the passive sensors (fusion), or without the passive sensors (if not available). The left column shows the average position error per radar update (4 seconds) for both conditions with the radar only (i.e., no passive sensors). The middle column shows the error when the radar measurements are fused with the passive sensor measurements.

Note the relative improvement (%) due to fusion in estimating target position shown in the right column.

Table 2. Position Estimation for Helicopter Example

	Radar Only	Radar + Passive Sensors	Percent Improvement
LOS unobstructed error (m)	60	39	35
LOS obstructed error (m)	474	188	60
Total error (m)	263	112	57

The multi-sensor tracker simulation results as shown in Figure 4 indicate that, on the average, the multi-sensor tracker tracked the maneuvering helicopter within 1.5 degrees from the truth for a total of 832 seconds about 92% of the time period of 904 seconds in which the Hind-D was within the sensors' detection range. For the IRST only, the corresponding numbers are 1.2 degrees, 546 seconds, 68%, and 804 seconds, respectively. During periods of no reports, the smaller prediction errors from the multi-sensor tracker significantly facilitate target reacquisition.

Bearing-Only Evaluation Analysis

The merits of multi-sensor tracking versus single-sensor tracking from the IRST, NI, and RFDF were evaluated in terms of bearing error against the known groundtruth. As shown for the Hind-D target in Table 3, the multi-sensor tracker had a total of 833 reports and 44 predictions. The individual sensors had fewer reports, ranging from 762 for the NI to 178 for the RFDF and had to make more predictions (as many as 202 for the IRST). These predictions are necessary when no reports are available from any sensors and the 3-D tracker requires a bearing reading.

Table 3. Bearing Tracker Performance Comparison for Helicopter (average error in degrees)

	IRST	NI	RFDF	All + Radar
Observed (number of contacts)	1.2 (316)	1.9 (762)	1.4 (178)	1.5 (833)
Prediction (number of predictions)	55.7 (202)	96.2 (49)	106.9 (139)	28.2 (44)
Total (tracker operations)	34.8 (518)	23.7 (811)	70.8 (317)	6.5 (877)

SUMMARY

This paper describes a sensor fusion simulation system and focuses on the merits of multi-sensor tracking. The system configuration chosen is based on a simple alpha-beta tracking approach. In this paper, we are primarily concerned with the tracking improvements that could result from fusing passive

sensors with a radar in a ground-to-air scenario. Our results indicate that intermediate angle information can, on the average, improve a 3-D radar's prediction of target location by more than a factor of two.

ACKNOWLEDGMENTS

We would like to acknowledge Dave Kabel for his technical contributions to this study. We would also like to thank Beverly Littlejohn for her assistance in preparing this manuscript.

REFERENCES

1. Automation in Combat Aircraft, National Academy Press, Washington, D.C. 1982.
2. Benedict, T.R. and Bordner, G.W. (1962); "Synthesis of an optimal set of track-while-scan smoothing equations," IRE Trans. on Automatic Control, Vol. AC-7, No. 4, (Jul), pp. 27-32.
3. Bhagavan, B. and Polge, R. (1974); "Performance of g-h filter for tracking maneuvering targets," IEEE Trans. on Aerosp. and Electron. Sys., Vol. AES-10, No. 6, (Nov), pp. 864-866.
4. Blackman, S.S. (1986); Multiple Target Tracking with Radar Applications, Artech House, Dedham, MA.
5. Gelb, A., (1974); Applied Optimal Estimation, Cambridge, MA: MIT Press.
6. Greer, T.H. (1985); "Artificial Intelligence: a new dimension in EW," Defense Electronics (Oct), pp. 108-128.
7. Quigley, A. (1973), "Tracking and associated problems", IEEE Conf. on Radar - Present and Future, London, (Oct), pp. 352-359.
8. Singer, R.A. and Benke, K. (1971), "Real-time tracking filter evaluation and selection for tactical applications," IEEE Trans. on Aerospace and Electronic Sys., Vol. AES-7, No. 1, (Jan), pp. 100-110.
9. Trunk, G.V. and Wilson, J.D. (1987), "Association of DF Bearing Measurements with Radar Tracks," IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-23, No. 4, (Jul), pp. 438-447.

ROBUST FUSION OF LOCATION INFORMATION

Raymond McKendall and Max Mintz

Department of Systems Engineering
and
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104-6389

Abstract

The purpose of this paper is to examine a sensor fusion problem for location data using statistical decision theory (SDT). The contribution of this paper is the application of SDT to obtain: (i) a robust test of the hypothesis that data from different sensors is consistent; and (ii) a robust procedure for combining the data which pass this preliminary consistency test. Here, robustness refers to the statistical effectiveness of the decision rules when the probability distributions of the observation noise and the a priori position information associated with the individual sensors are uncertain. Location data refers to observations of the form: $Z = \theta + V$, where V represents additive sensor noise and θ denotes the "sensed" parameter of interest to the observer. While the theory developed in this paper applies to several uncertainty classes, the focus of this presentation is on ϵ -contamination models, which allow one to account for heavy-tailed deviations from nominal sampling distributions.

ROBUST FUSION OF LOCATION INFORMATION

Raymond McKendall and Max Mintz

Department of Systems Engineering
and
Department of Computer and Information Science
University of Pennsylvania
Philadelphia, PA 19104-6389

1 Introduction

The purpose of this paper is to examine a sensor fusion problem for location data using statistical decision theory (SDT). The contribution of this paper is the application of SDT to obtain: (i) a robust test of the hypothesis that data from different sensors is consistent; and (ii) a robust procedure for combining the data which pass this preliminary consistency test. Here, robustness refers to the statistical effectiveness of the decision rules when the probability distributions of the observation noise and the a priori position information associated with the individual sensors are uncertain. Location data refers to observations of the form: $Z = \theta + V$, where V represents additive sensor noise and θ denotes the "sensed" parameter of interest to the observer. The parameter θ is referred to as a location parameter, since the distribution of Z is obtained from the distribution of V by a translation. This paper does not focus on a specific robotics problem, but seeks instead to abstract the essential components of a family of such data fusion problems and obtain a corresponding family of solutions. While the location parameter fusion problem is only one of many possible fusion paradigms, it does provide a useful starting point for considering more complicated problems, e.g., nonlinear sensor models of the form: $Z = h(\theta) + V$.

The fusion of location data for location estimation has been discussed in the robotics literature by Ayache and Faugeras (1987), Chatila and Laumond (1985), Durrant-Whyte (1986a, 1986b), Matthies and Shafer (1986), and Smith et al (1986, 1987). In Ayache and Faugeras (1987), Chatila and Laumond (1985), and Smith et al (1986, 1987), the authors assume the sensor noise can either be adequately modeled by Gaussian distributions with known means and covariances, or by distributions characterized only by specified first and second moments. In this latter situation the analysis is limited to affine decision rules which are evaluated on the basis of quadratic loss. Subject to these limitations (affine procedures and quadratic loss), the results of this decision model are "equivalent" to invoking a Gaussian model. Whereas, in Durrant-Whyte (1986a, 1986b) and Matthies and Shafer (1986), the authors recognize the need to address the existence of non-Gaussian sampling distributions. While Gaussian models offer a degree of mathematical elegance and simplicity, the adoption of Gaussian models for sampling distributions imposes

substantial risk on the decision-maker when the actual sampling distributions possess heavy tails, e.g., they exhibit departures from the Gaussian model in the form of ϵ -contamination uncertainty classes. This lack of robustness of decision procedures based on Gaussian sampling distributions has been discussed in the statistics literature for more than thirty years. Detailed examinations of the theory and applications of robust statistical inference appear in the monographs by Huber (1981) and Hampel et al (1986).

In the sequel we: (i) delineate several paradigms for robust fusion of multi-sensor location data; (ii) introduce some essential nomenclature and definitions from SDT; (iii) review the earlier decision-theoretic results on which this paper is based; and (iv) present and discuss a methodology for robust fusion of multi-sensor location data.

Our presentation emphasizes the statement and application of the relevant theory. Proofs of theorems are omitted. The reader is referred to journal articles and reports for these details.

2 Paradigms for Sensor Fusion of Location Data

In this section we delineate several paradigms for robust fusion of location data. We restrict our attention to observations of one-dimensional location parameters. The results of this one-dimensional analysis can be applied to the multidimensional case by doing a component by component analysis. Alternatively, one can pursue a formal multidimensional extension of the methodology presented in this paper.

The general one-dimensional paradigm is delineated as follows. We assume that we are given the sampled outputs of r sensor systems $\{S_i : 1 \leq i \leq r\}$. We denote the k^{th} sampled output of S_i , $1 \leq k \leq N_i$ by:

$$Z_{ik} = \mu_i + W_i + \theta_i + V_{ik}, \quad (2.1)$$

where:

- $a_i \leq \theta_i \leq b_i$, denotes an unknown location parameter with known bounds a_i and b_i . (Note: the bounds a_i and b_i may assume infinite values.) In many applications there is a

common interval of location parameter uncertainty for all sensors. However, there is no need to make this assumption in the following mathematical developments.

- μ_i , denotes a known constant (offset) associated with the position of sensor S_i with respect to a common origin.
- V_{ik} , denotes the additive observation noise associated with the k^{th} observation (sample) from S_i . The random variables $\{V_{ik} : 1 \leq k \leq N_i\}$ are assumed to be independent and identically distributed (i.i.d.). We further assume that the noise process associated with S_i is independent of the noise process associated with S_j , when $i \neq j$. Finally, we assume that the probability distribution of V_{ik} belongs to a given uncertainty class of distributions, \mathcal{F}_i . We do *not* assume that the noise processes associated with different sensors are identically distributed.
- W_i , denotes the uncertainty in the position of sensor S_i with respect to a common origin. We consider two cases: (i) the position uncertainty of S_i can be expressed by a known interval $[u_i, w_i]$ — with no a priori probabilistic description; or (ii) the position uncertainty of S_i can be expressed by an unknown probability distribution from a given uncertainty class \mathcal{P}_i . In each case, we assume that the position uncertainty of S_i is independent of the observation noise $\{V_{ik} : 1 \leq k \leq N_i\}$, and independent of the observation noise and position uncertainty of the other sensors.

Remark 2.1 Without loss of generality, we can assume that the known offsets $\{\mu_i : 1 \leq i \leq r\}$ are each zero, since nonzero values can be subtracted from the observations $\{Z_{ik}\}$. Further, if the known, generally asymmetric, interval of uncertainty $[a_i, b_i]$ in θ_i is finite, then the observations $\{Z_{ik}\}$ can be shifted and the interval of uncertainty $[a_i, b_i]$ can be replaced by $[-d_i, d_i]$, where $d_i = (b_i - a_i)/2$. Similarly, we can assume the interval of sensor position uncertainty (where applicable) is again symmetric. Thus, (2.1) can be replaced by:

$$Z_{ik} = W_i + \theta_i + V_{ik}, \quad (2.2)$$

where: $|\theta_i| \leq d_i$, and (where applicable) $|W_i| \leq \eta_i$, $1 \leq i \leq r$.

The uncertainty classes \mathcal{F}_i and (where applicable) \mathcal{P}_i , $1 \leq i \leq r$, denote neighborhoods in the space of probability distributions which are deemed to adequately characterize the uncertainty in the specifications of the sampling distributions. Models for several uncertainty classes are described in Sections 4, 5, and 6.

As stated in the introduction, the purpose of this paper is to examine a sensor fusion problem for location information using SDT. The contribution of this paper is the application of SDT to obtain: (i) a robust test of the hypothesis that data from different sensors is consistent, i.e., testing the hypothesis that $\theta_i = \theta_j$, $1 \leq i < j \leq r$; and (ii) a robust procedure for combining the data which pass this preliminary consistency test. Again, robustness refers to the statistical effectiveness of the decision rules when the probability distributions of the observation noise and the a priori position information of the individual sensors are uncertain.

In the following section, we introduce the notions of robust minimax decision rules, robust confidence procedures. These concepts provide the basis for the developments in the remainder of this paper.

3 Nomenclature and Definitions from SDT

The standard statement of a minimax location parameter estimation problem includes as given: a parameter space Ω ; a space of actions \mathcal{A} ; a loss function L defined on $\mathcal{A} \times \Omega$; and a CDF F . If the underlying CDF is imprecisely known, then this standard minimax decision model must be reformulated to account for this additional uncertainty. Statistical decision rules which are applicable in this more general problem setting are referred to as robust procedures.

This paper considers robust fixed size confidence procedures for a restricted parameter space. These robust confidence procedures are based, in turn, on the solution of a related robust minimax decision problem:

Let \mathbf{Z} denote a vector of N i.i.d. observations of a scalar random variable with CDF $F(z - \theta)$, where $F \in \mathcal{F}$, a given uncertainty class. Let $\Omega = \mathcal{A} = [-d, d]$, and define a zero-one loss function L on $\mathcal{A} \times \Omega$:

$$L(a, \theta) = \begin{cases} 0, & |a - \theta| \leq e; \\ 1, & |a - \theta| > e; \end{cases} \quad (3.1)$$

where $e > 0$, is given. Further, let $R(\delta, \theta, F) = E[L(\delta, \theta) | \theta, F]$ denote the risk function of the decision rule δ given $\theta \in \Omega$ and $F \in \mathcal{F}$.

Definition 3.1 An estimator δ^* is said to be a robust minimax estimator for θ , if for all δ :

$$\sup_{\substack{\theta \in \Omega \\ F \in \mathcal{F}}} R(\delta^*, \theta, F) \leq \sup_{\substack{\theta \in \Omega \\ F \in \mathcal{F}}} R(\delta, \theta, F).$$

Based on these definitions and assumptions, we seek a robust minimax estimator δ^* for θ . For brevity, we restrict our consideration to the case when d/e is an integer ≥ 2 .

Observation 3.1 The connection between the robust minimax rule $\delta^*(\mathbf{Z})$ and a robust fixed size confidence procedure is obtained by noting that:

$$C^*(\mathbf{Z}) = [\delta^*(\mathbf{Z}) - e, \delta^*(\mathbf{Z}) + e]$$

can be interpreted as a robust confidence procedure of size $2e$ which has the highest confidence coefficient $\inf_{\theta, F} P_{\theta, F}[\theta \in C^*(\mathbf{Z})]$.

Sections 4, 5, and 6 of this paper are organized as follows:

Section 4 reviews the solution to the related minimax estimation problem where $F = \mathcal{N}(0, \sigma^2)$, and σ is given. These results provide the basis for the solution to the robust minimax estimation problem where $\mathcal{F} = \{F = \mathcal{N}(0, \sigma^2) : \sigma \leq \sigma_u\}$.

Section 5 extends these robust minimax results to uncertainty classes which contain non-Gaussian, asymmetric, and discontinuous CDF's. Robust median-minimax rules are introduced and evaluated.

Section 6 develops a theory and methodology for robust sensor fusion of location information based on the theory presented in Sections 4 and 5.

4 Robust Minimax Rules and Gaussian Uncertainty Classes

4.1 Preliminary Minimax Results

Zeytinoglu and Mintz (1984) addresses the related minimax estimation problem where $F = \mathcal{N}(0, \sigma^2)$, and σ is given. The main result requires Definition 4.1 and is summarized by Theorem 4.1:

Definition 4.1 Let \mathcal{C} denote the class of nonrandomized, odd, monotone nondecreasing decision rules $\delta: E^1 \rightarrow \mathcal{A}$. Let $\Delta \subset \mathcal{C}$ denote the set of rules $\delta(t)$, defined for $t \geq 0$ by:

$$\delta(t) = \begin{cases} d - e, & c + a_n + 2ne \leq t; \\ \cdot \\ \cdot \\ \cdot \\ t - a_2, & c + a_2 + 2e \leq t < c + a_2 + 4e; \\ 2e + c, & c + a_1 + 2e \leq t < c + a_2 + 2e; \\ t - a_1, & c + a_1 \leq t < c + a_1 + 2e; \\ c, & c \leq t < c + a_1; \\ t, & 0 \leq t < c; \end{cases} \quad (4.1)$$

where: $0 \leq a_1 \leq a_2 \leq \dots \leq a_n < \infty$, $d = (2n + 1)e + c$, and c equals zero (e) if d is an odd (even) multiple of e . (Note: due to the existing symmetry, all function definitions are stated for nonnegative arguments.)

Theorem 4.1 Let L denote the loss function (3.1). If $Z \in \mathcal{N}(\theta, \sigma^2)$, where σ is given, then for any $N \geq 1$ there exists a (globally) minimax admissible rule $\delta^* \in \Delta$ which is Bayes with respect to a least favorable prior distribution λ^* . Further, δ^* depends on Z through the sample mean \bar{Z}_N .

Proof: See Zeytinoglu and Mintz (1984).

Observation 4.1 (Zeytinoglu and Mintz, 1984) If $N = 1$, $\delta \in \Delta$, and F is any continuous CDF which is symmetric about zero, then:

$$R(\delta, \theta, F) = \begin{cases} F(a_n - e), & d - 2e < \theta \leq d; \\ F(a_{n-1} - e), & \theta = d - 2e; \\ F(-a_n - e) + F(a_{n-1} - e), & d - 4e < \theta < d - 2e; \\ \vdots & \vdots \\ F(-a_2 - e) + F(a_1 - e), & c + e < \theta < c + 3e; \\ F(-a_2 - e) + (c/e)F(-e) \\ \quad + (1 - c/e)F(-a_1 - e), & \theta = c + e; \\ F(-a_1 - e) + (c/e)F(-e) \\ \quad + (1 - c/e)F(-a_1 - e), & 0 < \theta < c + e; \\ 2F(-a_1 - e), & \theta = 0. \end{cases} \quad (4.2)$$

Remark 4.1 $R(\delta, \theta, F)$, $R(\delta^*, \theta, F)$, and λ^* have the following characteristics:

- If $\delta \in \Delta$, then $R(\delta, \theta, F)$ is a piecewise constant function of θ over the sets of a finite partition of Ω .
- The minimax rules δ^* are "almost" equalizer rules, in the sense that the nondegenerate piecewise constant segments of the risk function are equalized at the minimax risk M .
- The risk expression (4.2) can be readily modified to include CDF's F which are both asymmetric and discontinuous. The generalized (asymmetric) risk function $R(\delta, \theta, F)$ is again a piecewise constant function of θ over the sets of the finite partition of Ω expressed in (4.2).
- λ^* is defined by a density function which is symmetric and piecewise constant.

4.2 Robust Minimax Rules

Definition 4.2 Let:

$$\mathcal{F} = \{F = \mathcal{N}(0, \sigma^2) : \sigma \leq \sigma_u\}, \quad (4.3)$$

denote an uncertainty class of Gaussian distributions.

Definition 4.3 The CDF $F_u = \mathcal{N}(0, \sigma_u^2)$ defines the upper-envelope of \mathcal{F} (4.3) in the sense that: $F(x) \leq F_u(x)$ for all $F \in \mathcal{F}$, and $x \leq 0$.

The following theorem, which is the main result of this section, extends the results of Theorem 4.1 to the robust minimax estimation problem.

Theorem 4.2 Let \mathcal{F} denote the uncertainty class (4.3) with upper-envelope $F_u = \mathcal{N}(0, \sigma_u^2)$. Let δ^* denote the minimax rule obtained through Theorem 4.1 based on a sample size N and CDF F_u . There exists a bound $B(d/e, N, F_u)$, such that if $e \geq B$, then δ^* is a robust minimax admissible Bayes rule in the sense of Definition 3.1. Further, δ^* depends on Z through \bar{Z}_N .

Proof: See Zeytinoglu and Mintz (1988).

Example 4.1 Let $d = 3e$, $e = 0.1$, $N = 49$, $\sigma_u = 2$, $F_u = \mathcal{N}(0, \sigma_u^2)$, and $G = \mathcal{N}(0, \sigma_u^2/N)$. Applying Theorem 4.2, the minimax rule $\delta^*(\bar{Z}_N)$ and risk function $R(\delta^*, \theta, F_u)$ corresponding to F_u are:

$$\delta^*(\bar{Z}_N) = \begin{cases} 2e, & a_1 + 2e \leq \bar{Z}_N; \\ \bar{Z}_N - a_1, & a_1 \leq \bar{Z}_N < a_1 + 2e; \\ 0, & 0 \leq \bar{Z}_N < a_1; \end{cases} \quad (4.4)$$

$$R(\delta^*, \theta, F_u) = \begin{cases} G(a_1 - e), & e < \theta \leq 3e; \\ G(-a_1 - e), & \theta = e; \\ 2G(-a_1 - e), & 0 \leq \theta < e; \end{cases} \quad (4.5)$$

where a_1 satisfies:

$$G(a_1 - e) = 2G(-a_1 - e). \quad (4.6)$$

In this example $a_1 = 0.092$, and the corresponding minimax risk is 0.49. The bound B , which is derived in Zeytinoglu and Mintz (1988), is:

$$B(d/e, N, F_u) = -(1/2\sqrt{N})F_u^{-1}(1/4) = 0.0966.$$

Here, $e \geq B$, and thus δ^* (4.4) is a robust minimax rule.

5 Robust Minimax Rules and Non-Gaussian Uncertainty Classes

5.1 Preliminaries

This section extends the robust minimax results of Section 4 to uncertainty classes which contain non-Gaussian, asymmetric, and discontinuous CDF's.

Definition 5.1 Let \mathcal{F} denote an uncertainty class with upper-envelope F_u :

$$\mathcal{F} = \{F: F(x^-) \leq F_u(x), x \leq 0; \text{ and } F(x) \geq F_u(x), x > 0\}, \quad (5.1)$$

where F_u has a density function which is unimodal and symmetric about zero. (Note: $F(x^-)$ denotes the left-hand limit.)

Remark 5.1 We allow F_u to be substochastic, i.e., F_u can have less than unit probability mass. Thus, the usual ϵ -contamination models can be represented by \mathcal{F} (5.1).

The main results of this section are based on the following theorem which addresses the existence and construction of \mathcal{C} -minimax and minimax rules for single-sample decision problems. (Note: A rule is (robust) \mathcal{D} -minimax if it is (robust) minimax within the class \mathcal{D} . A rule is \mathcal{D} -Bayes if it is Bayes within the class \mathcal{D} . A rule is \mathcal{D} -admissible if it is admissible within the class \mathcal{D} .)

Theorem 5.1 Let $N = 1$. If the CDF F has a density function which is unimodal and symmetric about zero, then there exists a \mathcal{C} -minimax rule $\delta^* \in \Delta$. Further, if F possesses a (strictly) monotone likelihood ratio, then δ^* is a minimax (admissible) Bayes rule.

Proof: See Zeytinoglu and Mintz (1984).

5.2 The Single-Sample Case

The following theorem extends the results of Theorem 5.1 to the single-sample robust \mathcal{C} -minimax estimation problem.

Theorem 5.2 Let $N = 1$, \mathcal{F} denote the uncertainty class (5.1) with upper-envelope F_u , and δ^* denote the \mathcal{C} -minimax rule obtained through Theorem 5.1 based on CDF F_u . There exists a bound $B(d/e, N = 1, F_u)$, such that if $e \geq B$, then δ^* is a robust \mathcal{C} -minimax rule. Further, if F_u possesses a (strictly) monotone likelihood ratio, then δ^* is a robust minimax (admissible) Bayes rule.

Proof: See Zeytinoglu and Mintz (1988).

Example 5.1 [an ϵ -contamination model] Let $d = 3e$, and \mathcal{F} denote the uncertainty class:

$$\mathcal{F} = \{F: F = (1 - \epsilon)\Phi + \epsilon H\}, \quad (5.2)$$

where: $\Phi = \mathcal{N}(0, 1)$, the CDF H is symmetric about zero, and $0 < \epsilon < 1/2$. The corresponding (substochastic) upper-envelope is:

$$F_u = (1 - \epsilon)\Phi + \epsilon/2. \quad (5.3)$$

In this example B is:

$$B(d/e, N = 1, F_u) = -(1/2)F_u^{-1}(1/4). \quad (5.4)$$

Applying Theorem 5.2, the \mathcal{C} -minimax rule $\delta^*(Z)$ and risk function $R(\delta^*, \theta, F_u)$ corresponding to F_u are:

$$\delta^*(Z) = \begin{cases} 2e, & a_1 + 2e \leq Z; \\ Z - a_1, & a_1 \leq Z < a_1 + 2e; \\ 0, & 0 \leq Z < a_1; \end{cases} \quad (5.5)$$

$$R(\delta^*, \theta, F_u) = \begin{cases} F_u(a_1 - e), & e < \theta \leq 3e; \\ F_u(-a_1 - e), & \theta = e; \\ 2F_u(-a_1 - e), & 0 \leq \theta < e; \end{cases} \quad (5.6)$$

where a_1 satisfies:

$$F_u(a_1 - e) = 2F_u(-a_1 - e), \quad (5.7)$$

or equivalently,

$$\Phi(a_1 - e) = 2\Phi(-a_1 - e) + e/2(1 - e). \quad (5.8)$$

Thus, if $e \geq B$, then δ^* (5.5) is a robust C -minimax rule for this ϵ -contamination model.

This solution is easily extended to other values of d/e and nominal distributions. The required calculations include: the computation of the vector a which parameterizes the underlying C -minimax rule δ^* , and the computation of the bound $B(d/e, N = 1, F_u)$ — which are each readily obtained by means of a Newton-Raphson algorithm.

5.3 The Multi-Sample Case

This section extends the robust C -minimax results of Theorem 5.2 to the multi-sample problem by restricting the class of estimators to rules of the form $\delta(T(Z))$, where: $\delta \in C$, $T: E^N \rightarrow E^1$, and $T(Z)$ possesses a density function which is unimodal and symmetric about θ . Examples of candidate T statistics include: the sample mean, the sample median, and other symmetric linear combinations of order statistics. In the remainder of this section we consider the sample median. See Zeytinoglu and Mintz (1988) for a comparison of the performance of these restricted decision rules with an alternative — the highest posterior density (HPD) credible set, Berger (1985).

Definition 5.2 Let Z_M denote the median of the N observations Z . The decision rule $\delta^*(Z_M)$, defined by the composition $\delta^* \circ Z_M$, is said to be a median-minimax estimator for θ , if δ^* is a minimax rule in the usual sense. The respective definitions of robust median-minimax rules, C -median-minimax rules, and robust C -median-minimax rules are obtained as before. (Note: If N is even, $Z_M = (Z_{[N/2]} + Z_{[(N/2)+1]})/2$.)

Theorem 5.3 Let $N > 1$. If the CDF F has a density function which is unimodal and symmetric about zero, then there exists a C -median-minimax rule $\delta^* \in \Delta$. Further, if the CDF of $(Z_M - \theta)$ possesses a (strictly) monotone likelihood ratio, then δ^* is a median-minimax (median-admissible) median-Bayes rule.

Proof: See Zeytinoglu and Mintz (1988).

Example 5.2 Let $d = 3e$, $e = 0.2133$, $N = 3$, and F denote the CDF of the double exponential distribution:

$$F(x) = \begin{cases} 1 - (1/2) \exp(-x), & x \geq 0; \\ (1/2) \exp(x), & x < 0. \end{cases} \quad (5.9)$$

Let F' denote the CDF of the centered sample median $Z_M - \theta = T(Z) - \theta$. In this example $F'(t) = F^2(t)(3 - 2F(t))$. Applying Theorem 5.3, the C -median-minimax rule $\delta^*(T)$ and risk function $R(\delta^*, \theta, F)$ are:

$$\delta^*(T) = \begin{cases} 2e, & a_1 + 2e \leq T; \\ T - a_1, & a_1 \leq T < a_1 + 2e; \\ 0, & 0 \leq T < a_1; \end{cases} \quad (5.10)$$

$$R(\delta^*, \theta, F) = \begin{cases} F'(a_1 - e), & e < \theta \leq 3e; \\ F'(-a_1 - e), & \theta = e; \\ 2F'(-a_1 - e), & 0 \leq \theta < e; \end{cases} \quad (5.11)$$

where a_1 satisfies:

$$F'(a_1 - e) = 2F'(-a_1 - e). \quad (5.12)$$

In this example $a_1 = e = 0.2133$, and the corresponding C -median-minimax risk is 0.50.

Theorem 5.4 Let $N > 1$, \mathcal{F} denote the uncertainty class (5.1) with upper-envelope F_u , and δ^* denote the C -median-minimax rule obtained through Theorem 5.3 based on CDF F_u . There exists a bound $B(d/e, N, F_u)$, such that if $e \geq B$, then δ^* is a robust C -median-minimax rule. Further, if the upper-envelope CDF of $(Z_M - \theta)$ possesses a (strictly) monotone likelihood ratio, then δ^* is a robust median-minimax (median-admissible) median-Bayes rule.

Proof: See Zeytinoglu and Mintz (1988).

Example 5.3 [Example 5.2 revisited and extended] Let $d = 3e$, $e = 0.2133$, $N = 3$, F_u denote the CDF of the double exponential distribution (5.9), and \mathcal{F} denote (5.1). In this example $B(d/e, N, F_u)$ is:

$$B(d/e, N, F_u) = -(1/2)G^{-1}(1/4) = e, \quad (5.13)$$

where: $G = F_u^2(3 - 2F_u)$. Since $e = B$, it follows from Theorem 5.4 that δ^* (5.10) is a robust C -median-minimax rule with corresponding maximum risk 0.50.

5.4 Two Special Cases

There are two limiting cases which are worthy of special mention:

Case 1. ($d = 2e$) If $d = 2e$, then Theorems 4.2, 5.2 and 5.4 apply for all $e > 0$, i.e., in each instance B is zero. The robust rules obtained in Theorems 4.2, 5.2, and 5.4 are respectively: Z_N , Z , and Z_M truncated to $[-e, e]$, with respective minimax risks: $F_u(-\sqrt{N}e)$, $F_u(-e)$, and $G_u(-e)$, where G_u denotes the CDF of the median based on F_u .

Case 2. ($d \rightarrow \infty$) If $\Omega = E^1$, then Theorems 4.2, 5.2 and 5.4 apply for all $e > 0$, i.e., in each instance B is again zero. However, in this case the resulting robust rules are extended Bayes. The robust rules obtained in Theorems 4.2, 5.2, and 5.4 are respectively: Z_N , Z , and Z_M , with respective minimax risks: $2F_u(-\sqrt{N}e)$, $2F_u(-e)$, and $2G_u(-e)$, where G_u again denotes the CDF of the median based on F_u .

These limiting cases provide useful upper and lower bounds for the minimax risk for intermediate values of d .

6 Robust Fusion of Location Information

6.1 Preliminary Remarks

In this section we develop a theory and methodology for robust fusion of multi-sensor location information based on Sections 4 and 5. Our approach contains two distinct phases:

- **Phase I** provides a test of the hypothesis that the location data (2.2) from sensor \mathcal{S}_i is consistent with the location data from sensor \mathcal{S}_j , where $i < j$, i.e., we test the hypothesis that $\theta_i = \theta_j$, $i < j$.
- **Phase II** provides a means of combining the location data from the individual data sets which "pass" the Phase I test, i.e., those deemed to be consistent.

In both phases of this process, we seek procedures which are robust to heavy-tailed deviations from the nominal sampling distribution, such as exhibited in ϵ -contamination uncertainty classes. Our usage of the terminology "robust" is also intended to imply that the procedures have satisfactory behavior when the actual sampling distribution coincides with the nominal, e.g., a given Gaussian distribution.

6.2 Sample Sizes and Uncertainty Classes

In developing suitable consistency tests, there are three classes of sample sizes to address: (i) the single sample case, $N_i = 1$; (ii) the small sample case, $1 < N_i \leq 20$; and (iii) the large sample case, $N_i > 20$. In defining these classes, it is important to observe that: (i) The transition ($N = 20$) between the small

sample and large sample cases is not a precise threshold value — the appropriate selection of this threshold is dependent on the uncertainty classes which define the given decision problem; and (ii) The sample sizes N_i and N_j can belong to different sample size domains.

The selection of appropriate sensor noise uncertainty classes $\{\mathcal{F}_i : 1 \leq i \leq r\}$ is an important issue in the development of a methodology for robust fusion of multi-sensor location information. Since, at the minimum, we seek to account for the occurrence of noise distributions with heavy tails, it is appropriate to consider ϵ -contamination uncertainty classes. In the sequel, we adopt a unimodal symmetric ϵ -contamination model \mathcal{F}_{ϵ_i} for each sensor \mathcal{S}_i , $1 \leq i \leq r$. In particular, we adopt the ϵ_i -contaminated Gaussian model for sensor \mathcal{S}_i which is defined by:

$$\mathcal{F}_{\epsilon_i} = \{F : F = (1 - \epsilon_i)\Phi + \epsilon_i H\}, \quad (6.1)$$

where: $\Phi = \mathcal{N}(0, 1)$, the CDF H is unimodal and symmetric about zero, and $0 < \epsilon_i < 1/2$.

It is also necessary to model the a priori position uncertainty in each sensor. For brevity, we restrict our attention to non-stochastic models of the form: $|W_i| \leq \eta_i$, $1 \leq i \leq r$, where $\eta_i \geq 0$ is given.

6.3 Phase I — Robust Consistency Tests

The following procedure provides a robust test of the hypothesis that $\theta_i = \theta_j$, $i < j$.

Case 1: ($\eta_i = 0$) Let \mathcal{M}_i , $1 \leq i \leq r$, denote the class of CDF's defined by the centered sample median Z_{M_i} of N_i i.i.d. samples with CDF $F \in \mathcal{F}_{\epsilon_i}$ (6.1). Let \mathcal{M}_{ij} , $1 \leq i < j \leq r$, denote the class of CDF's defined by the difference of the centered sample medians $(Z_{M_i} - \theta_i) - (Z_{M_j} - \theta_j)$, where the CDF's of the centered sample medians $(Z_{M_i} - \theta_i)$ and $(Z_{M_j} - \theta_j)$ belong, respectively, to \mathcal{M}_i and \mathcal{M}_j . It follows from these definitions that the class \mathcal{M}_{ij} is a set of symmetric unimodal distributions. Further,

$$Z_{M_i} - Z_{M_j} = \theta_i - \theta_j + \nu_{ij}, \quad (6.2)$$

where: the CDF of ν_{ij} belongs to \mathcal{M}_{ij} ; and the a priori uncertainty in $\theta_i - \theta_j$ is given by the interval $[-d_{ij}, d_{ij}]$, where $d_{ij} = d_i + d_j$.

Hence, we can construct a robust fixed size ($2e$) confidence procedure for $\theta_i - \theta_j$. The parameter e is selected by the decision maker and denotes his tolerance to small errors between θ_i and θ_j . The desired procedure $[\delta^* - e, \delta^* + e]$ is obtained via Theorems 5.2 and 5.4. Finally, the test of the hypothesis $\theta_i = \theta_j$ is obtained as follows: we reject $\theta_i = \theta_j$ if $0 \notin [\delta^* - e, \delta^* + e]$. From this test we also obtain the probability that $\theta_i - \theta_j \in [\delta^* - e, \delta^* + e]$. Examples of applications of this class of robust consistency tests appears in McKendall and Mintz (1988).

Case 2: ($\eta_i > 0$) If the uncertainties η_i , $1 \leq i \leq r$, are suitably small, then the previous test can be applied with only a small modification in the definition of the uncertainty class \mathcal{M}_{ij} . The details appear in McKendall and Mintz (1988).

6.4 Phase II — Robust Fusion of Consistent Multi-Sensor Location Information

The following procedure provides a robust estimate of the common location parameter θ of r sensor data sets, $r \geq 3$. We observe at the outset that, when V_1 and V_2 possess very heavy tails, it is generally not useful to attempt to combine only two observations of the form:

$$Z_1 = \theta + V_1$$

$$Z_2 = \theta + V_2.$$

For example, if V_1 and V_2 are independent Cauchy $C(0, 1)$ random variables, then any convex combination of Z_1 and Z_2 will be a $C(0, 1)$ random variable. Further, there are random variables with continuous unimodal symmetric density functions whose sample mean, for any sample size $N > 1$, has greater variability than any of its N i.i.d. components. (See for example, Brown

and Tukey (1946).)

Case 1: ($\eta_i = 0$) Let $\{Z_{M_i} : 1 \leq i \leq r\}$ denote the sample medians of r consistent data sets with common location parameter θ . In order to simplify the exposition, we further assume that the r sample medians are identically distributed. Let Z_{M_A} denote the median of the $\{Z_{M_i} : 1 \leq i \leq r\}$. Let \mathcal{M}_A denote the uncertainty class of the centered sample median $Z_{M_A} - \theta$. Each CDF $F \in \mathcal{M}_A$ is unimodal and symmetric about zero. Thus, we can apply Theorem 5.2 to obtain a robust fixed size confidence procedure $[\delta^* - e, \delta^* + e]$ for θ . Examples of applications of this class of confidence procedures for the robust fusion of consistent multi-sensor location information appears in McKendall and Mintz (1988).

Case 2: ($\eta_i > 0$) If the uncertainties η_i , $1 \leq i \leq r$, are suitably small, then the previous robust confidence procedure for estimating θ can be applied with only a small modification in the definition of the uncertainty class \mathcal{M}_A . The details appear in McKendall and Mintz (1988).

6.5 Other Research Issues and Further Results

One of the sufficiency conditions in the theory of robust fixed size confidence procedures delineated in Sections 4 and 5 of this paper is that the value of $e \geq B$ — a given bound. The delineation of a complete theory requires that we also consider the case where $e < B$. It has been shown, by Martin and Mintz (1985), that where $e < B$, it is necessary to consider randomized decision rules to obtain robust fixed size confidence procedures. Further results on the robust fusion of location information pertaining to asymptotic theory, α -trimmed means, randomized decision rules, and nonmonotone decision rules appear in McKendall and Mintz (1988).

References

- Ayache, N. and Faugeras, O. D. (1987). Maintaining representations of the environment of a mobile robot. *Proceedings of the fourth international symposium on robotics research*. Santa Cruz, California 109-121.
- Berger, J. O. (1985). *Statistical Decision Theory and Bayesian Analysis*. Second Edition. Springer-Verlag, New York.
- Bickel, P. J. (1965). On some robust estimates of location. *Ann. Math. Statist.* 36 847-858.
- Brown, G. W. and Tukey, J. W. (1946). Some distributions of Sample Means. *Ann. Math. Statist.* 17 1-12.
- Chatila, R. and Laumond, J-P. (1985). Position referencing and consistent world modeling for mobile robots. *Proceeding of the 1985 IEEE International Conference on Robotics and Automation*. 138-145.
- Durrant-Whyte, H. F. (1986a). *Integration, Coordination, and Control of Multi-Sensor Robot Systems*. PhD Dissertation. University of Pennsylvania, Philadelphia, Pennsylvania.
- Durrant-Whyte, H. F. (1986b). Consistent integration and propagation of disparate sensor observations. *Proceeding of the 1986 IEEE International Conference on Robotics and Automation*. 1464-1469.
- Gastwirth, J. L. and Rubin, H. (1969). On robust linear estimates. *Ann. Math. Statist.* 40 24-39.
- Hampel, F. R., Ronchetti, E. M., Rousseeuw, P. J., and Stahel, W. A. (1986). *Robust Statistics: The Approach Based on Influence Functions*. Wiley, New York.
- Huber, P. J. (1981). *Robust Statistics*. Wiley, New York.
- Martin, K. and Mintz, M. (1985). Randomized robust confidence procedures. *Chapter 4 of Final Report on Contract F33615-83-C-3000*. Submitted to the Air Force Flight Dynamics Laboratory, Wright-Patterson AFB, OH.
- Matthies, L. and Shafer, S. A. (1986). Error modeling in stereo navigation. *Technical Report CMU-CS-86-140*. Com-

puter Science Department, Carnegie-Mellon University, Pittsburgh, Pennsylvania.

McKendall, R. and Mintz, M. (1988). Robust fusion of location information. *GRASP Laboratory Technical Report*. University of Pennsylvania, Philadelphia, Pennsylvania.

Smith, R., Self, M., and Cheeseman, P. (1986). Estimating uncertain spatial relationships in robotics. *Proceedings of the Workshop on Uncertainty in Artificial Intelligence*. Philadelphia, Pennsylvania 267-287.

Smith, R. and Cheeseman, P. (1987). On the representation and estimation of spatial uncertainty. *International Jour. Robotics Research*. 5(4) 56-68.

Zeytinoglu, M. and Mintz, M. (1984). Optimal fixed size confidence procedures for a restricted parameter space. *Ann. Statist.* 12 945-957.

Zeytinoglu, M. and Mintz, M. (1988). Robust fixed size confidence procedures for a restricted parameter space. *Ann. Statist.* 16 To appear.

Acknowledgements: This research was supported in part by NSF/DCR 8410771, NSF DMC-8411879 and DMC-12838, US Air Force F49620-85-K-0018, US Air Force F33615-83-C-3000, US Air Force F33615-86-C-3610, DARPA/ONR, ARMY/DAAG-29-84-K-0061, NSF-CER DCR82-19196 A02, NIH NS-10939-11 as part of the Cerebrovascular Research Center, by DEC Corp., and LORD Corp.

THIS PAGE LEFT BLANK INTENTIONALLY

A MULTI-FREQUENCY MULTI-SENSOR DATA FUSION SYSTEM

Dr. Joseph J. Fitchek, Naval Ocean Systems Center (1)
John P. Lee, ORINCON (2)
Dr. David Herring, ORINCON (2)

The Prototype Information Correlation Exploitation System (PICES) is an Office of Naval Technology 6.2 level exploratory development program for an advanced tracker/correlator system and is managed through the Integrated Ocean Surveillance Block at the Naval Ocean Systems Center. This program addresses the needs of both the Navy's afloat and ashore ocean surveillance nodes and the needs of the surveillance community in general to correlate and fuse multi-frequency, multi-sensor contact report data for the detection, tracking, and classification of multiple surface and airborne targets. PICES effectively addresses surveillance operator overload due to the heavy decision processes and manual interactions associated with sensor data correlation, sensor ambiguity resolution and fusion, and target track generation, by largely automating these surveillance functions. This paper described PICES' features and presents results generated from data inputs at both the contact report and raw sensor output level.

PICES is a component of a research, development, test and evaluation (RTD&E) system using a multiple hypothesis tracker/correlator algorithm. The system is running at NOSC and is implemented on a VAX 11/785 with TEKTRONIX equipment for graphical display. The RTD&E system consists of simulated and real multi-sensor data drivers, fusion algorithms, Kalman filtering for target state updating, various target kinematic models, a customized database management system, measures of performance, and a powerful user query and real time graphical display interface.

The result of PICES development is an integrated tracking, data correlation, and multi-frequency, multi-sensor data fusion system that automatically generates surveillance scene hypotheses (possibilities due to information ambiguities) and ranks them on the basis of all sensor information available to the system at a given time. The system has demonstrated a powerful capability to automatically fuse a variety of contact report data types resulting in great alleviation of operator workload. PICES handles the following information types: unique attributes, geospatial, range and bearing, and ELINT parameters. The query and display capability allows very effective operator interaction with the system. The operator can generate graphical track displays, track overlays (for example, comparing ground truth with hypothesis predictions) and symbology along with hypothesis, target, and sensor information. In addition, the operator can use his surveillance experience to override and correct system track predictions and surveillance data inputs, when necessary.

The paper discusses PICES operation at the sensor output level on real HICAMP infrared imagery. Multi-band data was fused to track weak airborne target signals in high noise backgrounds, where single band tracking could not be performed. Finally, we discuss PICES brassboard prototype microcomputer implementation for potential transition to fleet operational use and describe future program direction.

- (1) Code 743, San Diego, CA, 92152, (619) 553-2486
- (2) 9363 Towne Centre Drive, San Diego, CA 92121, (619) 455-5530

A MULTI-FREQUENCY, MULTI-SENSOR, DATA FUSION SYSTEM

Dr. Joseph J. Fitchek, Naval Ocean Systems Center
John P. Lee, ORINCON Corporation
Dr. David Herring, ORINCON Corporation

1.0 INTRODUCTION

The Navy and other DOD services are faced with ever increasing surveillance operational requirements. These increasing surveillance requirements are driven by a number of factors. One of the most important is the need to maintain an ever-increasing area of coverage, in order to counter the threat posed by more powerful hostile missiles and aircraft and support over-the-horizon targeting of our own weapon assets. Furthermore, the need for new and improved sensor systems and communication channels creates an information explosion for the surveillance system. The general requirements of increased timeliness, accuracy, and throughput for surveillance product generation demands that the sensor data correlation and fusion and multi-target track generation processes be highly efficient, largely automated processes. Therefore, a critical aspect of evolving surveillance systems is the ability to fuse data from different sensor systems in order to generate scene descriptions which are high in information content and low in ambiguity.

The Prototype Information Correlation Exploitation System (PICES) is an Office of Naval Technology 6.2 level exploratory development program for an advanced tracker/correlator system and is managed through the Integrated Ocean Surveillance Block at the Naval Ocean Systems Center. This program addresses both the needs of the Navy's afloat and ashore ocean surveillance nodes and the needs of the surveillance community in general, to correlate and fuse multi-frequency, multi-sensor report data for the detection, tracking, and classification of multiple surface and airborne targets. PICES is a multi-hypothesis computer based tracker/correlator system with a powerful operator interface. PICES effectively addresses surveillance operator overload by largely automating the surveillance functions. This paper describes PICES and presents results generated from data inputs at both the contact report and raw sensor output level.

2.0 BENEFITS

In this section we describe some of the advantages anticipated as a result of replacing traditional manual tracking and correlation with an automated multi-hypothesis tracker/correlator system such as PICES.

2.1 Relieve Surveillance Operator Overload

If an operator is attempting to perform the tracking and correlation functions manually, he may become overwhelmed by the sheer volume of reports, resulting in reports being disregarded, or improperly correlated with existing tracks. In addition, when tracking a large number of targets, an operator may find it very difficult to make correlation decisions due to the number of variables which must be considered when making decisions, such as attribute mismatches, geospatial proximity, and ELINT parametric correlation.

Furthermore, given highly ambiguous data, an operator may be forced to delay making firm correlation decisions until further data is received, meaning he must not only consider the new data arriving, but must also keep in mind the impact of the new data on previous tentative correlations. The use of a computer based correlation system can relieve operator workload and greatly increase throughput in processing large amounts of data.

2.2 Improved Correlation and Tracking

In addition to its ability to process a large quantity of data, PICES is capable of providing a higher quality of output than can be provided by manual processing, or by automatic processing with less powerful algorithms. Accurate tracking is achieved by the use of a Kalman filter. Improved correlation performance results from the multi-hypothesis approach. Also, the retention of multiple views of a surveillance scene can provide valuable information in a tactical situation concerning the potential number and location of hostile forces.

3.0 APPLICATIONS

PICES has applications in any situation where decisions are being made as to the partitioning of an incoming ambiguous data set into groups of related measurements. In the following paragraphs we describe several potential application areas for PICES.

3.1 Ocean Surveillance

The original purpose for creating the multi-hypothesis system was to apply the PICES technology to sensor data correlation problems in ocean surveillance. These problems include wide area ocean surveillance and anti-submarine warfare.

3.2 Targeting

PICES can be used in weapons targeting by providing target localization information, in the form of a target latitude, longitude, and position uncertainty ellipse. In addition, PICES has a built in hit probability function which calculates the likelihood of hitting a selected target with a weapon. This hit probability can be used in a tactical situation to aid in the weapon release decision.

3.3 Multi-Frequency, Multi-Sensor Data Fusion

The current work addresses the correlation and fusion of sensor contact reports (i.e. ocean surveillance products) for target track generation. These reports have been generated by surveillance operators and are therefore evaluated sensor contacts. Our approach can also be adopted to the fusing of unevaluated multi-frequency, multi-sensor data. This data for fusion can be in a variety of forms: multi-spectral infrared and radar imagery,

radar parameters such as range, doppler, bearing, or bearings only information such as that received from acoustic, IRST, and ESM sensors. Also, multi-sensor attribute information can be incorporated in the fusion process. The fusing of unevaluated sensor data poses an interesting tracking problem, particularly in the case of dim target tracking, where target detection takes place only at the expense of a high number of false alarms. An example of using PICES for the fusion of multi-spectral infrared imagery data for target tracking with a moderate number of false alarms is discussed in Section 6.3.

3.4 Sensor Systems Evaluation

As a tracking and correlation testbed, PICES can provide data useful in the design and evaluation of sensor systems. Using real or simulated input data, the performance of different sensor mixtures and sensor parameters can be measured. For example, PICES could be used to test the effectiveness of combining data from a highly geographically accurate sensor which does not provide attributes, with a sensor which is less positionally accurate, but carries some attribute information. Measures of performance generated by PICES can form the basis of comparison between two sensor configurations.

4.0 PICES TESTBED HARDWARE/SOFTWARE

Figure 1 shows a block diagram of the current PICES hardware configuration. The central computer responsible for performing the Kalman filtering, hypothesis management, and database query processing is a DEC VAX 11/785 running the VMS operating system. The present PICES hardware system resides in a NOSC research and development environment. Contact reports are stored in a disk file and are read in sequentially and processed by the PICES algorithm. Any graphical plots which the user has requested are automatically updated after each report has been processed. In order to minimize the impact on processing speed due to the updating of the graphics display, a TEKTRONIX 4132 Unix workstation is used to offload graphics processing from the VAX. The VAX sends the minimal amount of information needed to update the display over the RS 232 interface to the TEKTRONIX 4132. The 4132 then expands the command and communicates the changes to the 4129 display terminal over a high speed DMA interface. The use of parallel graphics processing has allowed PICES to maintain its real time graphical output with minimal impact on the speed of processing input reports.

Further processing power is available on the 4129 graphics terminal, which allows the operator to pan and zoom the display region without intervention from the 4132 or the VAX. The 4129 also has a dialog area where status information and database query responses are displayed.

Figure 2 shows a block diagram of the major software components of the PICES testbed. At the heart of the system is the data fusion algorithm, which is being fed from data generated by the scenario generator (or other data source). The user interacts with PICES through the query language and graphics interface. Scenarios can be constructed based on an analysis of available ocean surveillance products (OSP). The KBES block refers to a knowledge based expert system decision aid which is not yet part of PICES, but may be incorporated in the future.

5.0 SYSTEM OVERVIEW

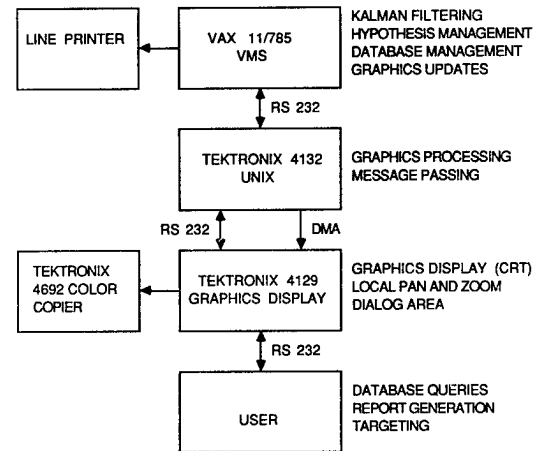
In this section we describe important PICES features and testbed components shown in Figure 2.

5.1 Kalman Filter Tracker

PICES uses an extended Kalman filter tracker. The Kalman filter state variables are target latitude and longitude, along with the corresponding velocities. Either a straight line or an Integrated Ornstein Uhlenbeck (IOU) maneuvering target motion model may be chosen at run time. Input reports may contain position estimates in the form of a latitude, longitude, and error ellipse, or in the form of a bearing wedge with minimum and maximum range. Both types of reports may be mixed in the same scenario.

5.2 Hypothesis Management

Given a collection of target position reports, there are a large number of ways in which the reports may be grouped together to form tracks. Each possible grouping, or partition, of the input reports into tracks is called a hypothesis. PICES is an example of a multi-hypothesis tracker, which means that more than one correlation hypothesis is retained after processing a contact report.



PICES HARDWARE BLOCK DIAGRAM

FIGURE 1

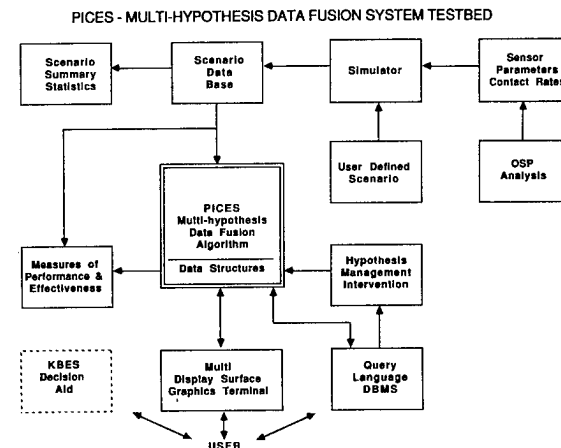


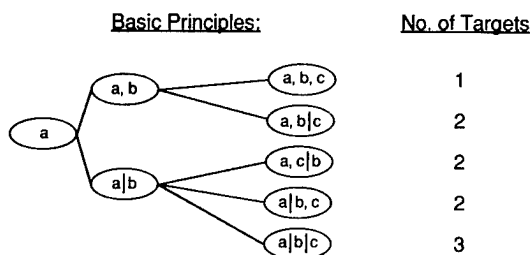
FIGURE 2

Figure 3 illustrates the formation of the hypothesis "tree". Reports hypothesized to be from a single target are separated by commas, while those reports hypothesized to be from different targets are separated by vertical lines. The base of the tree consists of the single report "a". When report "b" is received, two hypotheses can be formed. The upper branch represents the hypothesis that reports "a" and "b" are from the same target, while the lower branch represents the hypothesis that the reports are from two distinct targets. The receipt of report "c" results in a total of 5 possible hypotheses. It is easily seen that if all possible hypotheses are retained, the hypothesis tree will rapidly grow to an unmanageable size as more reports are processed. Thus, low scoring hypotheses are pruned so that the number of hypotheses retained after processing a contact report remains roughly constant.

The advantage of maintaining multiple hypotheses is that decisions concerning ambiguous reports may be delayed until further data is available. An example where this approach is effective is during track initiation. When the first report on a target has been received, it is generally impossible to estimate its velocity. Typically, a velocity of 0 is assigned to the track. When another report arrives near the previous report, it may not yet be clear whether this report is from the same target as seen before, or is from a new target. A multi-hypothesis tracker would therefore form two hypotheses concerning the origin of the second report. The hypothesis that the second report is from the same target as the first would allow the velocity of the target to be estimated. Now, if a third report comes in, its position can be compared to the position expected from a target moving with the velocity calculated from the previous hypothesis. If the positions match closely, we now have evidence that the previous hypothesis linking the first two reports is correct. The scoring algorithm would then give the single target hypothesis a higher score than the hypothesis that all reports are from different targets. As more reports are received from the target, the single target hypothesis would continue to grow in probability.

When an input report is received, a number of correlation hypotheses are generated and scored. Prior to formation of hypotheses, the incoming report is compared with each of the existing tracks in the database for geospatial, attribute, and ELINT parametric feasibility. This step, known as gating, eliminates candidate tracks which have virtually no probability of associating with the incoming report and helps keep the number of hypotheses which must be scored and maintained to a manageable level.

MULTI-HYPOTHESIS DATA FUSION



Assume three unequated contact reports. Five hypotheses ranging from 1 target to 3 targets are generated. False alarms are ignored in this example.

FIGURE 3

Following the gating step, a score is calculated for each of the possible associations of the incoming report with existing tracks, along with the probability that the report is from a new target. If the incoming report contains ELINT parameter data as well as geospatial data, the association score reflects both types of data.

5.3 Operator Interface

A number of operator interface features have been incorporated into PICES, to make it more productive as a research testbed, and to ultimately allow it to make the transition to an operational setting. The following paragraphs describe some of the operator interface features of PICES.

Database Management and Query System

One of the most powerful and useful aspects of the PICES user interface is the database query language. Using a syntax similar to the commands used to operate commercial relational database systems, the operator is able to call up for tabular display any information stored in the PICES database. The information most often accessed is the hypothesis list and the list of tracks contained in the top hypothesis. The user may specify which fields in the database record he wishes to see, and may place conditions on the fields, or sort the displayed records according to a field. For example, to see a list of the tracks (referred to in PICES as "clusters") contained in the best (most probable) hypothesis which are known to be flying a Soviet flag, the user may issue a display command as follows:

```
DISPLAY CLUSTERS FROM HYPOTHESIS MAXIMUM FOR FLAG EQ SOVIET
```

As shown in Figure 4, PICES responds with a list of all the items from the database which satisfy the conditions. To save on keystrokes, the above command could be shortened by using three or four character abbreviations.

In addition to the ability to query the database, the user is also provided with an editing capability. For example, if an operator has knowledge of a ship's identity which is not available to PICES, he may edit the name field in the ship's database record to fill in the missing information. The operator may also delete hypotheses or tracks if he believes PICES is in error. We felt that the editing capability was important so PICES can work with the human operator, who, although slow compared to the speed of a computer, may have knowledge and experience which the tracker software does not possess.

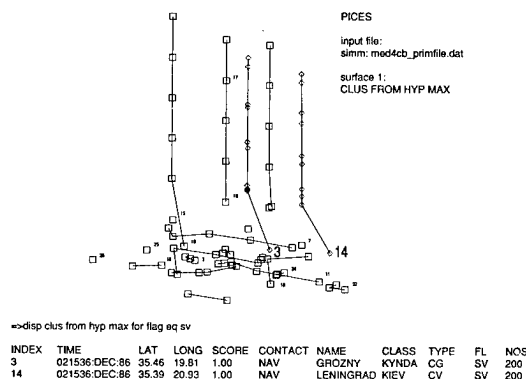


FIGURE 4

Real Time Graphics Display

The most important user interface component present in PICES is the graphical display. The user may request plots of the tracks being formed using the full power of the query language to place conditions on the displayed data. As reports are processed, the tracks on the display are automatically updated to reflect the new data. For example, the operator may wish to restrict his attention to a particular geographic region, and may be interested only in naval vessels. By issuing a command to PICES, the conditions would be evaluated for each track in the scenario, and only those tracks satisfying the conditions would be displayed.

This graphics capability, which would be required in an operational setting, has also proved invaluable during the research and development phase of the project. Changes to the algorithm which affect performance can be instantly seen as a result of the visual feedback provided by the graphics display. For example, the ground truth tracks may be overlaid with the tracks calculated by PICES for comparison. The data which may be plotted include the raw sensor measurements, the ground truth tracks generated by the simulator, and the tracks corresponding to the hypotheses formed by the PICES algorithm.

Figures 4 and 5 illustrate some of the graphics capabilities of PICES. Figure 4 shows a view of a scenario where five vessels moving south are entering a shipping lane. The user may request plots of targets of interest, as in Figure 5 where the user has asked for a display of only the targets known to be Soviet.

Targeting Figure of Merit

An interesting feature which was added to PICES was the calculation of a hit probability. The operator may select a target and a search ellipse, and have PICES calculate the probability of hitting the target with a missile capable of searching the region inside the ellipse centered at the estimated location of the target of interest. This hit probability takes into account the possibility that the target of interest may not be in the search region, as well as the possibility that another target may be inside the region and may be hit instead of the intended target. In a threat situation at sea, this capability could provide a commander with valuable information which would help

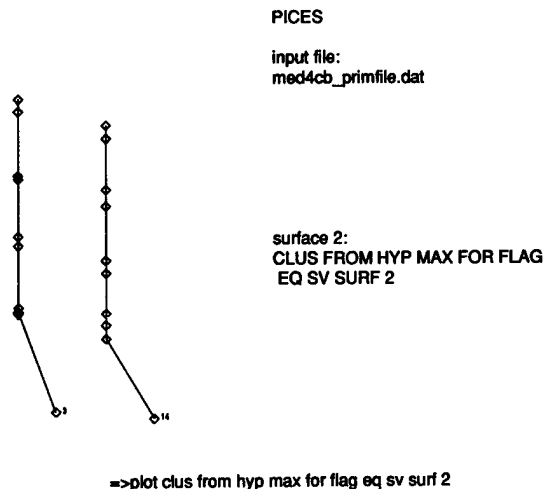


FIGURE 5

him decide whether a weapon should be released or not.

5.4 Performance Measures and Input Data Ambiguity

A number of performance measures have been incorporated into PICES to aid in quantifying the tracking and correlating capabilities of the algorithm, as well as evaluating the effectiveness of different sensor characteristics. The performance measures may be called up at any time during the processing of a simulated data set. Included in the performance measures are several quantities which measure the ambiguity present in the input data set. The input ambiguity is important to know, since performance of the algorithm is always relative to the difficulty of the scenario being tested. Work on quantifying the ambiguity of the input data set is continuing. At this time, the following have been implemented:

Hypothesis Ratio

The hypothesis ratio is one of several measures of the ambiguity of the input data set. It is calculated as the ratio of the number of hypotheses which could be formed after the gating step to the number of hypotheses which would have been formed in the absence of gating. Unambiguous data, for example data in which the targets are widely separated geographically or have unique attributes, will show a low hypothesis ratio, as most of the potential associations will be prevented during the gating phase of the algorithm. On the other hand, for highly ambiguous data, gating will provide a less dramatic reduction in the number of candidate tracks, resulting in a large hypothesis ratio.

Hypothesis Evolution

Following the processing of a scenario, a plot of the ancestry of any selected hypothesis may be obtained. For example, the best hypotheses present at the end of the scenario may be selected for analysis, resulting in a plot of the ranking of all of its ancestor hypotheses in the hypothesis tree. This is another measure of the ambiguity of the input data, since highly ambiguous data should result in considerable fluctuation in the ranking of the hypotheses, while for unambiguous data the top ranking hypothesis would generally yield a descendant which is also the top ranking hypothesis.

Entropy

As a measure of correlational performance, one may consider the track entropy, defined by the following summation:

$$-\sum (A(i,j)/N) * \log(A(i,j)/NGT(j))$$

where

$A(i,j)$ = number of reports from ground truth target j which were assigned by the tracker to constructed track i ,

N = total number of reports processed, and

$NGT(j)$ = number of reports from ground truth track j .

The entropy gives a measure of the dispersion of ground truth tracks to tracks constructed by PICES. An entropy of 0.0 implies that each ground truth target has had all of its reports correctly assigned to a single constructed track. Large entropies reflect essentially a random assignment of ground truth reports to constructed tracks, indicating poor correlation performance. Entropy

based calculations applied to the input data set are currently being investigated as a measure of the inherent ambiguity of the scenario.

5.5 Simulator

In order to provide controlled scenarios for testing the tracking and correlating algorithms in PICES, a scenario simulator was developed. Scenarios with multiple targets and multiple sensor types may be created. For example, sensors may be modelled as active or passive, with the passive sensors only able to detect emitting targets. ELINT sensors, which detect both geolocation and ELINT parameters such as pulse repetition interval (PRI) and scan rate (SCAN) may also be modelled. Up to 5 emitters may be placed on each target, and changes in an emitter's PRI may be programmed to occur at any time during the course of the scenario. By turning all emitters on a target off, emissions control (EMCON) may be simulated.

Sensors may be selected to produce latitude/longitude or bearings only type of reports, and can be assigned different probabilities of detection and different reporting rates. Reports include error ellipse information and may be offset from the ground truth location. Target attributes may be given individual probabilities of detection, making it possible to simulate sensors with any mix of attribute data. Construction of realistic scenarios has been aided by analysis performed on real-world data, revealing typical data rates, measurement uncertainties, and availability and types of attribute information on incoming reports.

6.0 EXPERIMENTAL RESULTS

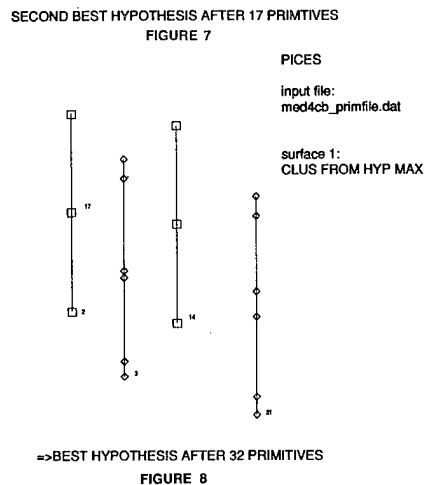
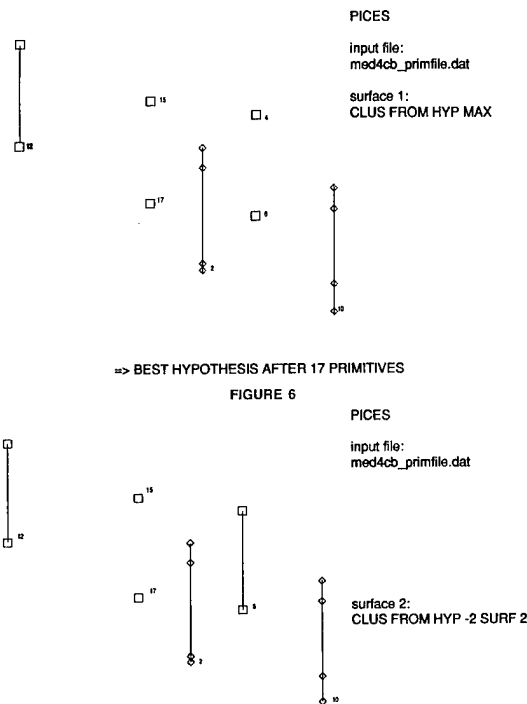
In this section, we describe some experimental results obtained by running PICES on several types of scenarios. In addition, we illustrate many of the operator interface features described earlier.

6.1 Soviet Attack Force

A scenario was created which had 5 Soviet warships entering a busy shipping lane in the Mediterranean, travelling in the shipping lane for two days, and emerging in an attack force. Figures 6 through 8 show the five Soviets entering the shipping lane at the start of the scenario, and are good examples of the multi-hypothesis approach at work. The Soviets enter the scene from the north, and travel in parallel formation until they reach the shipping lane. At that time, they turn southeast into the shipping lane. Figure 6 shows that PICES has incorrectly failed to link the reports from two of the vessels in the best hypothesis. However, as shown in Figure 7, the correct linkage for the second track from the right is retained in the second best hypothesis. The track in this hypothesis will carry a non-zero velocity, making it possible to accurately predict the location of the third report from the target. After more reports have been received, the correct hypothesis for the second track from the right is brought up to the top by the scoring algorithm, as shown in Figure 8. This hypothesis, since it did such a good job of predicting the third report from the vessel, is rewarded with a large probability, bringing it up from the second most likely hypothesis to the best hypothesis. PICES has also connected the reports from the second vessel from the left. Thus PICES has, in effect, corrected its earlier mistakes, corrections which would have been difficult if multiple hypotheses were not retained.

This scenario, in addition to being used to illustrate the value of a multi-hypothesis approach, was also used to test the effectiveness of mixing a

variety of sensors in order to investigate the feasibility of tracking targets in a high density shipping lane. One space based sensor, which gave position information without attributes, was included with two land based sensors, which reported some attribute information along with position. The Soviet vessels went into EMCON as they entered the shipping lane. A number of scenarios were generated with different values of reporting rates, measurement errors, and availability of attributes. As a result of these studies conclusions were drawn regarding the effectiveness of different sensor characteristics and sensor combinations.



6.2 ELINT Scenarios

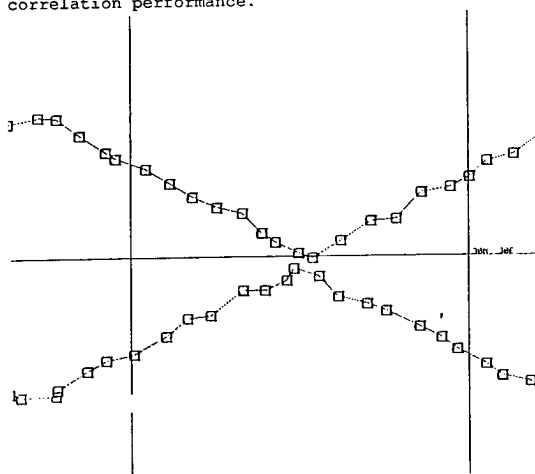
A number of scenarios were run to test the value of incorporating ELINT scoring into the algorithm. The scenarios constructed were based on the scenarios used to evaluate the POST (Prototype Ocean Surveillance Terminal) and TEC (Tactical ELint Correlator) ELINT correlators [Ref. 1] in an OTH-T

context. These are prototype research and development systems. In this reference, it was suggested that POST and TEC could be improved by incorporating geospositional scoring into the correlation decisions. Since PICES can use both geospositional and ELINT parameters in the hypothesis scoring process, it was a good candidate for testing that suggestion.

Two-Target Scenarios

Several two target scenarios, similar to the ones used in the POST/TEC evaluation, were tested using geospositional only, and using both geospositional and ELINT scoring. In general, we found that the performance of PICES on the two-target ELINT scenarios using geospositional data-only in the scoring algorithm was at least as good or better than the performance of either POST or TEC, while the performance of PICES with both geospositional and ELINT parameter scoring seemed to be significantly better. Since it was not possible to run PICES on exactly the same data set as was used to test POST and TEC, it is somewhat unfair to come to hard conclusions about the performance of PICES relative to the other trackers. However, it can be concluded that PICES performed quite well on these tests.

Figure 9 shows a close up of a recoiling target scenario, illustrating the close proximity of the targets at their closest point. PICES ran the entire scenario without making a single correlation mistake, using both geospositional and ELINT scoring. The ELINT parameters of the two targets were set about two standard deviations apart. Without using ELINT scoring, PICES would have tended to show the tracks crossing rather than recoiling, due to the straight line prediction of the Kalman filter. Thus, in the case of this scenario, the use of ELINT scores made a dramatic improvement in the correlation performance.



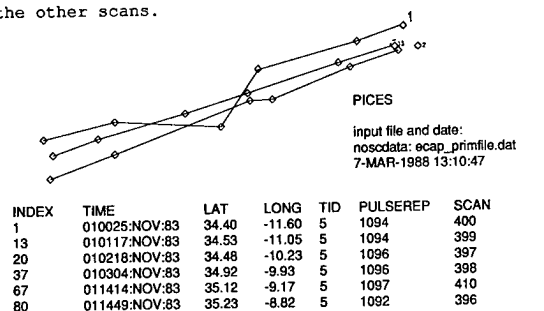
=> RECOILING TARGETS - 10 NM SEPARATION IN GEO,
2 STD DEV IN ELINT
FIGURE 9

ECAP Scenario

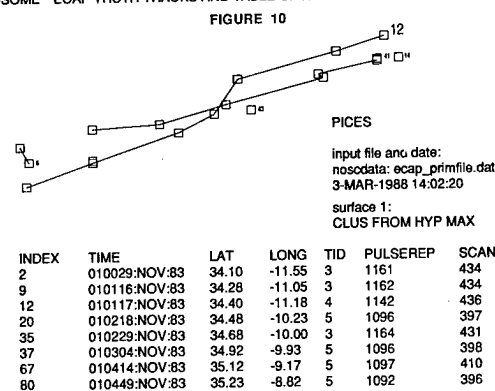
In order to test PICES on a large, realistic ELINT scenario a copy of the ECAP II scenario was used as input to PICES. The ECAP (ELINT Correlation Analysis Project) scenario is a large scenario containing six Soviet vessels and about 117 other vessels in the Strait of Gibraltar. The six Soviets engage in a search and rescue mission, with three ships arriving from the Atlantic and three from the Mediterranean. The contact reports contain both geospositional and realistic ELINT parametric data.

PICES was run on the ECAP scenario two times, first with ELINT scoring disabled, and then with ELINT scoring enabled. After 100 contact reports had been received, the scenario was paused and the measures of performance were calculated. The entropy was found to be more than twice as large for the case where ELINT scoring was not used than for the ELINT scoring case. Since large entropy reflects greater "confusion", we concluded that the use of ELINT scoring greatly aided the algorithm in making the correct correlation decisions.

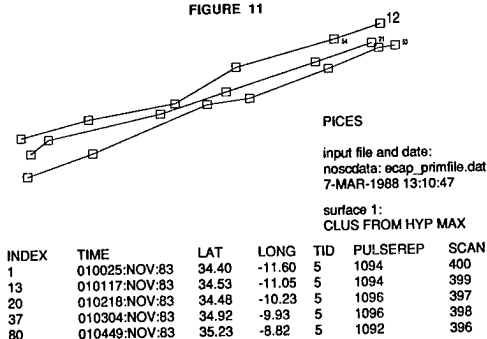
Figures 10, 11, and 12 further illustrate the benefits of ELINT scoring. In the following figures, values for PRI and SCAN have been altered so as not to reveal classified information. Figure 10 is a close up of the raw measurement data from three Soviets travelling east, showing the ground truth linkage of their contact reports. The tracks show a great deal of positional ambiguity in the measurement data, with the uppermost track crossing over the other two tracks. The table in the figure illustrates the output of a query to the database manager concerning the reports contained in the uppermost track. Note that the reported scan of 410 for one of the reports differs significantly from the other scans.



=> SOME ECAP TRUTH TRACKS AND TABLE OF REPORTS FROM TRUTH TRACK 1



=> ELINT SCORING DISABLED - CONTACT REPORTS FROM TRACK 12
FIGURE 11



=> ELINT SCORING ENABLED - REPORTS FROM TRACK 12
FIGURE 12

Figure 11 shows the tracks formed by PICES when tracking was based on geoposition only, with ELINT tracking disabled. Many correlation mistakes have been made, as can be seen by the "TID", or ground truth target id, column in the table for the uppermost track. Three different targets have been linked together here as a result of the large geopositional ambiguity.

Finally, Figure 12 shows PICES tracking the three Soviets using ELINT and geopositional scoring. The three targets are correlated perfectly with the exception of the report containing the outlying scan value, which is not linked with the other reports. This report, with track id 54 in the figure, will eventually be deleted, as it will fail to be updated by subsequent reports.

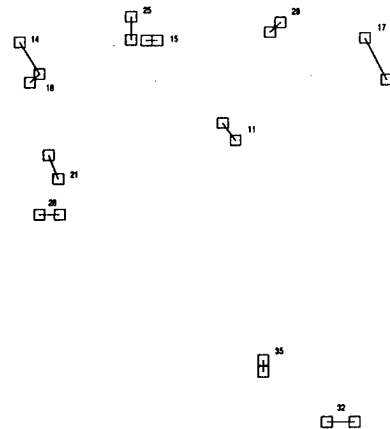
6.3 HICAMP Infrared Multi-Spectral Imagery Fusion

An experiment was conducted using HICAMP infrared imagery to see if the PICES algorithm could be used for low signature target detection. This situation differs significantly from the ocean surveillance scenarios described previously due to the extremely high data rate, and the presence of a large number of false alarms. A data set containing simultaneously measured short and long wave infrared imagery was obtained. Two targets were present, an F14 flying parallel to a drone. To present a more challenging problem to PICES, the drone, which was much dimmer than the F14, was chosen for tracking.

The IR data consists of about 100 frames of pixel amplitudes, corresponding to the energy observed in the sensor element during the duration of the frame. Several preprocessing steps were necessary before the data could be input to PICES. First, successive frames were differenced to remove stationary background values. Next, a peak detection step was applied to the data to be sure that any target detections were confined to a single pixel. Finally, to reduce noise, the data was thresholded by retaining only those pixels which exceeded a threshold. Any pixel which remained with a non zero value after the thresholding was converted to a position report and input to PICES. A typical frame of data consisted of about 10 noise values (false alarms). A target detection would occur about every 4 or 5 frames.

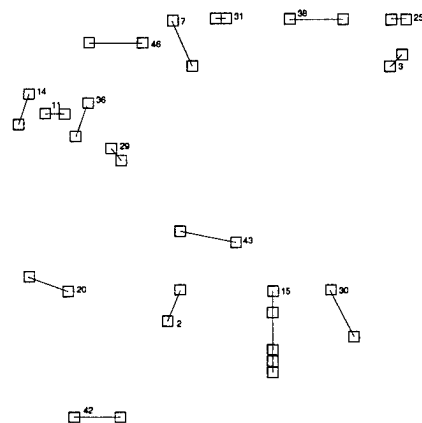
Figure 13 shows the candidate tracks being formed by PICES just after the drone has entered the scene in the long wave data set. The tracks formed by noise points tend to be short, and can usually be purged as a result of not being updated often enough, or because of velocities which would not be characteristic of a target of interest.

Figure 14 shows the beginning of a track for the drone showing up in the lower right quadrant of the figure after the target has been in the picture for several frames. Finally, Figure 15 shows a clear track being formed for the drone. Notice in comparing the three figures that noise tracks tend to come and go, and are held to a relatively constant, and manageable, level. In practice, some decision criteria will be employed to determine if a track represents a detection. The table in Figure 15 contains columns for velocity (LA_VEL and LO_VEL), average report to track association probability (ASSOC_R), and total number of reports in a track (TOTALH). These quantities can be used by PICES to make deletion decisions concerning false tracks. The drone in the above data set, while not entirely invisible to the trained eye in the raw data, was quite difficult to see, so that the ability of PICES to detect and track it was significant.



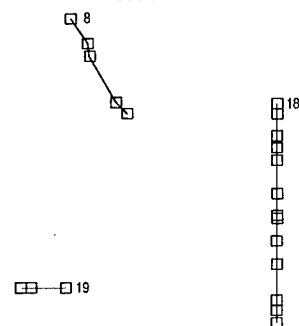
TRACKING INFRARED DATA

FIGURE 13



TARGET APPEARS AS CLUSTER 15

FIGURE 14



=>disp clus from hyp max for totalh get 2 sort totalh

INDEX	LAT	LONG	LA_VEL	LO_VEL	ASSOC_R	TOTALH
19	23.97	22.26	0.00	7.30	0.801	3
8	29.36	22.34	4.10	-2.82	0.794	5
18	27.66	26.53	7.68	0.00	0.949	13

FIGURE 15

In addition to the work done in tracking the drone in the long wave data, some experimentation was done on fusion of the short and long wavelength data sets. Unfortunately, upon close examination of the short wave data set, it appeared that the drone was not present, while the F4 was easily visible in both data sets. Therefore, to experiment with multiband fusion, we artificially dimmed the F4 in

the two data sets by randomly deleting detections until the target could not be seen or tracked in either data set alone. The two data sets were then merged and input to PICES, which was able to gather enough information in the fused data set to track the target. This example resulted in the fusion of individual band target signal to noise ratios of approximately 2 dB to yield an enhanced target track signal to noise ratio of 5 dB. The algorithm has not been fine tuned for this multi-frequency imagery problem, so that further work is needed to quantify the limitations of the algorithm for higher false alarm rates and lower target signal.

7.0 CURRENT WORK

In this section, we briefly mention the current work on the PICES project. One of the major thrusts of the current effort is to transition PICES from a research and development environment into an operational setting.

Microcomputer Implementation

An Intel 80386 based microcomputer has been purchased as a platform for the PICES algorithm. The computer is supplied with a high performance graphics board, a 19 inch color monitor, and a numeric coprocessor. We anticipate performance comparable to the performance we have obtained with the combination of the VAX, TEKTRONIX 4132 workstation, and TEKTRONIX 4129 graphics terminal, at a fraction of the cost. The move to put PICES on a microcomputer is aimed at making it a low cost, portable system for use in an operational environment.

Live Sensor Report Data Interface

A link to real time data will be provided for PICES by connecting PICES to the an ocean surveillance product report communications interface. This will move us further toward our goal of an operational home, and will be a source of real data for analysis and refinement of the tracking and hypothesis generation algorithms.

Live Sensor Report Data Analysis

A large amount of data gathered from the communications link will be analyzed for such characteristics as report rate, report accuracy and report content. The data gathered from this study

will help us create more realistic scenarios, and may suggest improvements to the algorithm.

8.0 FUTURE WORK

A major goal of future work on PICES will be to transition the system to a Navy operational setting. Some of the issues which need to be addressed include the possible need for higher computation speeds, an improved user interface, and fine tuning of the tracking and scoring methods. In addition, steps must be taken to ensure that PICES is compatible with existing data communications interfaces. As new high performance computer hardware, such as transputer boards or array processors, becomes available its possible use in the tracker/correlator system will be studied.

An additional application area for the PICES technology is in the area of data fusion. Further work is needed to identify algorithmic and hardware limitations in addressing low signal to noise target tracking and the associated high false alarm rate. The application of the multi-hypothesis approach to the tracking of targets characterized by low imagery contrast and low radar cross section (RCS) using a variety of sensor types would be an interesting multi-frequency, multi-sensor fusion problem.

9.0 CONCLUSIONS

The work done to date on PICES has resulted in a tracker/correlator system which includes many of the recent advances in tracker/correlator algorithms, as well as a powerful user interface. The effectiveness of the multi-hypothesis approach has been demonstrated in a research environment. Future work, including processing of live ocean surveillance sensor report data, will allow us to evaluate the performance of the microcomputer implementation of PICES on realistic scenarios. The next challenge is to gain operational experience, and use the knowledge gained to suggest improvements to the algorithms and user interface.

10.0 REFERENCES

1. Naval Ocean Systems Center, Johns Hopkins University/Applied Physics Laboratory, Naval Research Laboratory, "ELINT Correlator Evaluation", 1986. (SECRET).

THIS PAGE LEFT BLANK INTENTIONALLY

APPLIED PROBABILISTIC DATA ASSOCIATION

Jeff Brandstadt
(315) 793-7357

GE Aerospace
Aerospace Electronic Systems
Utica, NY 13503

ABSTRACT

Although significantly more complex than simple gating association and "nearest-neighbor" correlation techniques, Probabilistic Data Association (PDA) offers advantages in some applications. This paper examines two applications which illustrate the improvements that PDA techniques can offer. The first example applies PDA to the multispectral, kinematic tracking problem. Traditional techniques for correlating data from multiple sensors are insufficient if the target density is high and if the sensors have very different measurement accuracies. One sensor may detect and establish tracks on multiple targets while a sensor with lower resolution may detect and track only one target. The question is, which of the multiple tracks should benefit from the other sensor's measurement? PDA offers a method to resolve this question and also to aid in the formation and deletion of track files.

The second example applies PDA to the problem of correctly correlating attribute measurements (i.e., emitter type, engine type, probability Friend/Foe/Neutral, etc.) with kinematic data. Many times sensors that measure a target's attributes give only coarse location data; for example, an electronic support measure (ESM) system may supply high confidence target identification (ID), but only coarse angular accuracy. If there are a number of radar tracks within that angular window, which radar track should we tag with the ESM-generated ID? PDA offers a solution to this Attribute-Kinematic correlation problem.

The paper describes how to modify traditional fusion algorithms (nearest-neighbor correlation, Kalman filter kinematic fusion, and Bayesian attribute fusion) to produce an algorithm that will solve the two problems discussed above. It also discusses a method for determining the probabilities required to implement Probabilistic Data Association.

APPLIED PROBABILISTIC DATA ASSOCIATION

Jeff Brandstadt

GE Aerospace
Aerospace Electronic Systems
Utica, NY 13503

1.0 INTRODUCTION

GE Aerospace Electronic Systems, located in Utica, New York, designs only airborne sensors. We operate under strict size, weight, and speed constraints, and consequently attempt to keep the complexity of our processing to a minimum. It is not surprising, then, that we did not immediately embrace the recent advances in target tracking: Probabilistic Data Association (PDA), Joint Probabilistic Data Association (JPDA), Multiple Modeling, etc. These algorithms' complexity seemed to outweigh their benefits. As we looked at the problem more closely and as our computational capability increased, we began to see applications of these probabilistic techniques to our airborne problems. We feel that the multiple model filter improves our single-target fire control tracking accuracy, and we are investigating the possibility of replacing our simpler maneuver model filters with the more accurate, more complex multiple model technique. The following paper, however, discusses the application

of a variation of probabilistic data association to the surveillance/multiple-target track problems. We have identified two applications of PDA where we feel the benefits may outweigh the increase in complexity.

The first and perhaps the most significant benefit occurs when we apply PDA to the problem of correlating coarse ESM measurements to accurate radar tracks in the surveillance problem (see Figure 1).

Given that a surveillance radar has detected and begun to track a number of targets and the ESM system detects an emitter at a certain bearing with a large degree of angular uncertainty, the question is, which, if any, of the radar targets generated the detected emission? Traditional correlation techniques (simple gating and/or Chi-Square/Nearest-Neighbor) would tag the ESM measurement to the most likely target, the target closest to the reported bearing. In fact, it is highly probable that some other target within the beam generated the emission. Tagging the ESM measurement to the incorrect radar track results in an incorrect ID. Once

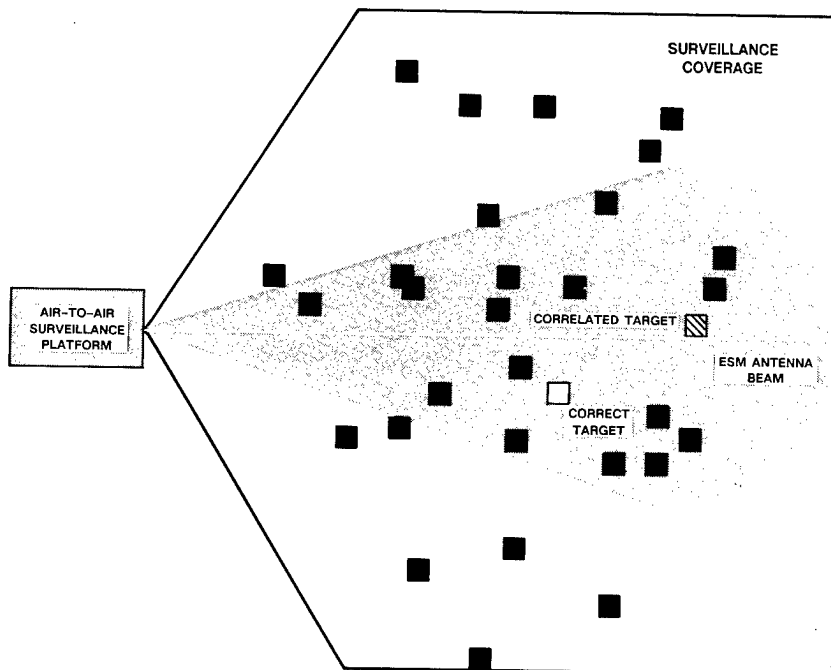


Figure 1. ESM-to-Radar Correlation

we have incorrectly identified a particular target, it is very difficult to correct the mistake without complicated and extensive logic. Because of the coarseness of the ESM system's bearing measurements, the emitter ID itself is used in the correlation process. If we identified a particular radar track as a specific emitter and we receive ESM reports of that same emitter, then those reports are immediately correlated to the identified track. This process actually reinforces incorrect ESM correlations and could result in high confidence, but completely incorrect identification. Thus, we must exercise caution when correlating ID information to as yet unidentified tracks. The Probabilistic Data Association algorithm exercises the required caution. It allows us to probabilistically correlate a single ID measurement to a number of tracks. Then over time the correct track receives the majority of the ID measurement information, and we can correctly identify the track with a high degree of confidence.

Another application in which the benefits of PDA may outweigh its computational burden is the multiple-track and raid cell assessment problem. Figure 2 illustrates a generic tracking situation where PDA offers improved performance. Assume that our multisensor platform is operating in a low-probability-of-intercept mode, and our primary surveillance sensor is an Infrared Search and Track (IRST). The IR system detects one target and initializes a track. To improve the range accuracy of the track, we cue the radar system to briefly track the target. Where the IR system detected only one target, the radar system now resolves three. Once the radar has provided range for the three targets, it breaks lock and reverts back to passive operation. The composite tracks on each of the three targets are now very accurate in both range and angle. As the targets close, the IR sensor continues to detect only one target. Without extra logic, the Chi-Square/Nearest Neighbor association algorithm will correlate the single IR measurement to only the most likely of the three tracks. It is a good possibility that the other two tracks will be terminated due to lack of data. When the targets approach closely enough for the IR system to resolve them, the multisensor system must reinitialize the two dropped tracks. Because the only measurements available for these two new tracks are infrared azimuth and elevation, their range estimates will be very poor. So poor, in fact, that they may not be included in the same raid cell!

Probabilistic Data Association offers a solution to this problem that is an alternative to solving the problem with a logic tree that continually assesses the number of targets associated with each measurement. Once multiple tracks are established, PDA distributes a single measurement to multiple tracks. In our simple example, where all three tracks are equally likely to have generated the single IR measurement, each track will probabilistically receive a portion of the measurement. No tracks will be terminated due to lack of data, and as a result, when the IR sensor finally resolves the targets, they will still be in the same raid cell with fairly accurate range estimates.

We feel that these benefits alone warrant a closer look at probabilistic data association techniques. The next step is to develop an algorithm that allows us to apply these techniques to the ESM/Radar correlation problem and to the multiple track/raid cell problem.

2.0 PROBABILISTIC CORRELATION OF ONE MEASUREMENT TO MULTIPLE TRACKS (PCOMMT)

Probabilistic Data Association assumes one target of interest and multiple measurements. Joint PDA assumes multiple targets and multiple measurements. [Ref. 1] For our applications, it is very difficult to have two or more measurements valid at exactly the same time. As a result, it is possible to sequentially order the measurements in time and process them individually. Then our association problem becomes the probabilistic correlation of one measurement to the multiple tracks already established. Figure 3 illustrates the problem. We must correlate the single measurement to the most likely track or tracks that generated it. We need to develop an algorithm that will allow us to probabilistically correlate one measurement to multiple tracks (PCOMMT). Once the correlation is complete, we must fuse the kinematic and attribute data in the measurement vector to the correlated track file. Figure 4 illustrates the necessary procedure given a one-measurement-to-one-track file correlation scheme. The Association/Correlation function selects the most likely measurement-to-track correlation. The Kinematic Fusion function employs a Kalman filter to fuse the kinematic portion of the measurement vector to the kinematic portion of the correlated track file. The Attribute Fusion function then fuses the remaining attribute portion of the measurement vector to the attribute portion of the state vector. (At GE Utica, we apply a Bayesian algorithm to execute the attribute fusion.) In our problems, we want to correlate one measurement to multiple tracks. Each one of the functions pictured in Figure 4 requires modification.

The Association/Correlation function requires the least modification of all the functions pictured. Figure 5 illustrates a more detailed block diagram of the Association/Correlation algorithm. After passing a series of gating tests, the candidate targets' "statistical distances" from the measurement vector are calculated. The track whose statistical distance is a minimum is the track that most likely generated the measurement and it is the track that is said "to correlate" with the measurement vector. If we want to correlate the measurement with multiple tracks, then we select a small number of tracks that most likely could have generated the measurement. That is, the correlated tracks are those three to ten tracks that have the smallest Chi-Square numbers or statistical distances to the measurement vector. Thus, to change a Nearest-Neighbor correlation algorithm to a PDA-type algorithm, we need only to allow more than one

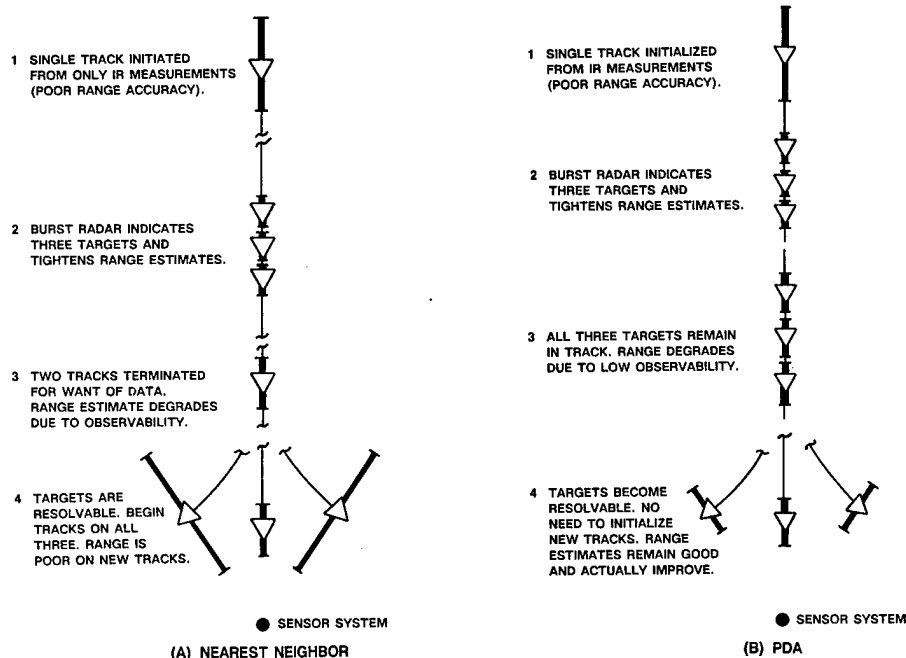


Figure 2. Ambiguous Target Track

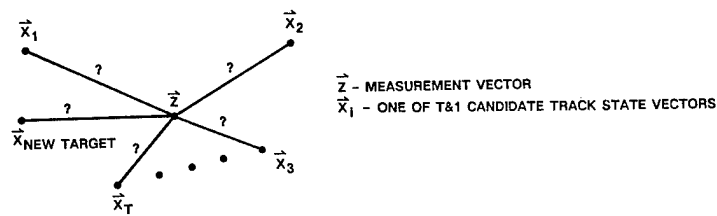


Figure 3. Which Track Vector(s) Should \vec{z} Update?

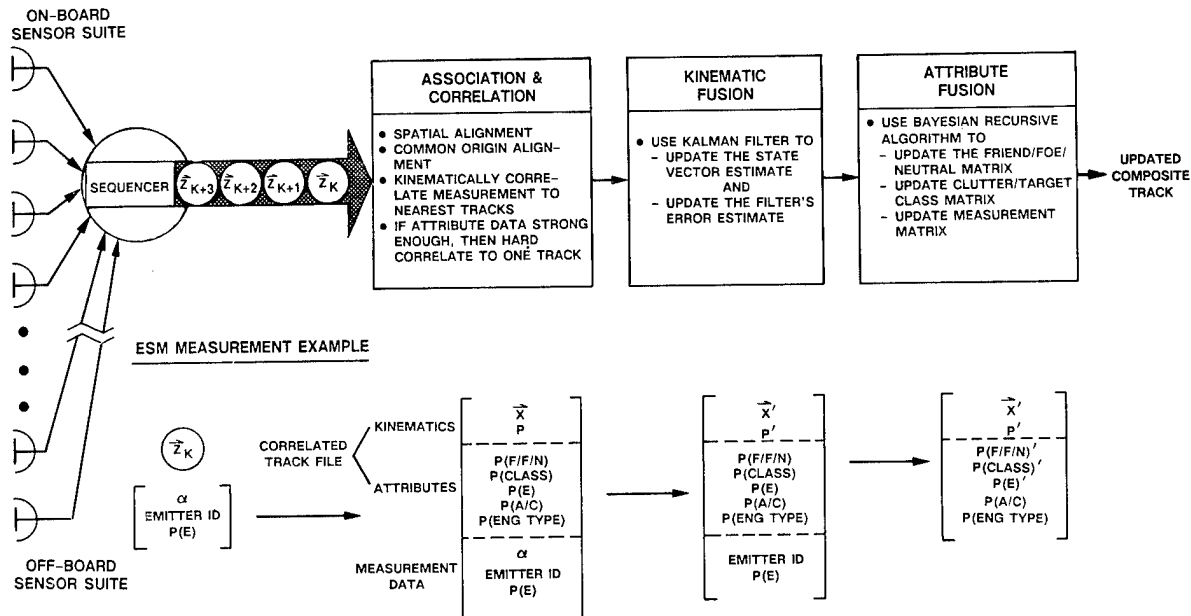


Figure 4. Functional Diagram of Measurement-to-Track Fusion

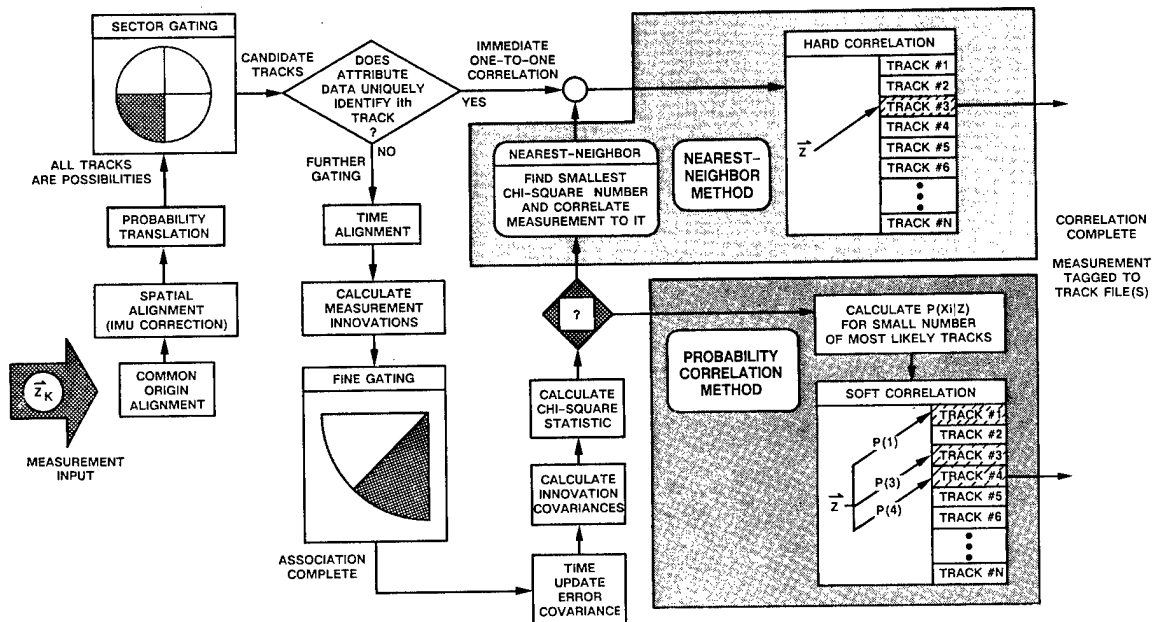


Figure 5. Association/Correlation Block Diagram

2.1 Modification to Kinematic Fusion

$$\hat{\vec{x}}_i(k/k-1) = \Phi \bullet \hat{\vec{x}}_i(k-1/k-1) + B \bullet \vec{U}(k-1) \quad (1)$$

$$P_i(k/k-1) = \Phi \bullet P_i(k-1/k-1) \bullet \Phi^T + Q_i(k-1) \quad (2)$$

$$K_i(k) = P_i(k/k-1) \cdot H_i^T \cdot [H_i \cdot P_i(k/k-1) \cdot H_i^T + R(k)]^{-1} \quad (3)$$

$$P_i(k/k) = [I - K_i(k) \bullet H_i] \bullet P_i(k/k-1) \quad (4)$$

$$\hat{\hat{X}}_i(k/k) = \hat{\hat{X}}_i(k/k-1) + K_i(k) \bullet [\vec{Z}(k) - H_i \bullet \hat{\hat{X}}_i(k/k-1)] \quad (5)$$

where $\hat{\mathbf{x}}_i(k/k)$ is the kinematic state vector estimate of target i valid at time t_k given data up to and including data valid at t_k . $\hat{\mathbf{x}}_i(k/k-1)$ is the estimate of the same state vector, but only using data up to time t_{k-1} . Φ is the state transition matrix; $\mathbf{B}\vec{\mathbf{u}}(k-1)$ is the deterministic forcing func-

If we know that track i did not generate the current measurement, $\vec{Z}(k)$, then we should update its estimate, $\hat{X}_i(k-1/k-1)$, to time t_k with equation 1; i.e.,

$$\hat{\vec{X}}_i'(k/k-1) = \Phi \bullet \hat{\vec{X}}_i(k-1/k-1) + B \bullet \vec{U}(k-1) \quad (6)$$

where we have included the prime (') on \hat{X} to indicate that this is an intermediate result, and where ϕ spans the time from t_{k-1} to t_k . But if track i did generate the measurement, then we should correct this predicted estimate with equation 5, i.e.,

$$\hat{\bar{X}}_i'(k/k) = \hat{\bar{X}}_i'(k/k-1) + K_i(k) \bullet [\bar{Z}(k) - H_i \hat{\bar{X}}_i'(k/k-1)]. \quad (7)$$

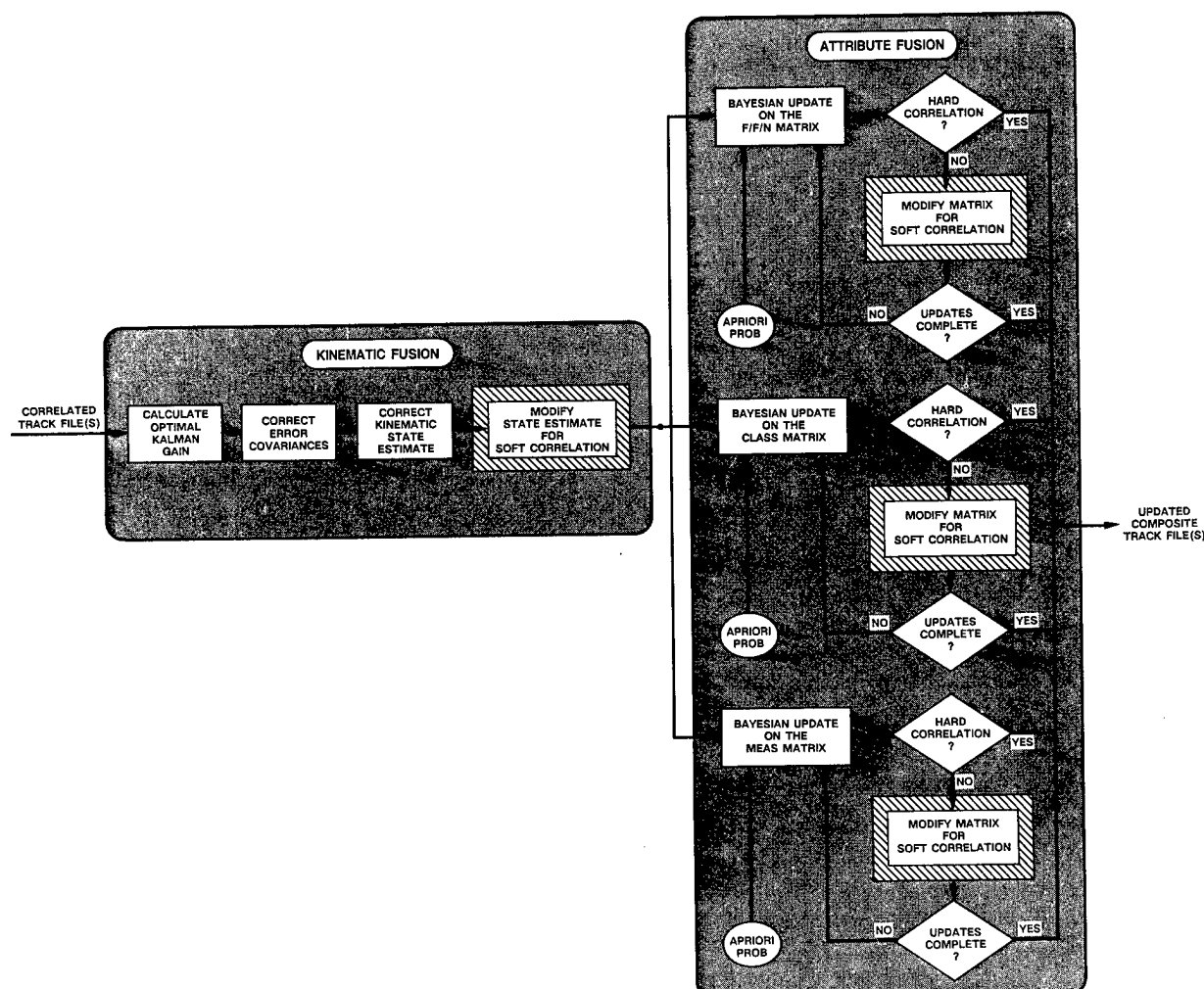


Figure 6. Kinematic and Attribute Fusion Modifications

We do not know, however, which case is true. Therefore, let us treat each case separately and then combine them probabilistically. Assuming for a moment that given the measurement $\vec{Z}(k)$, we know the probability that the i th track generated it, $P(\vec{X}_i/\vec{Z})$, we can combine the two update possibilities (equations 6 and 7) with the Total Probability Theorem.

$$\hat{\vec{X}}_i(k) = \hat{\vec{X}}_i'(k/k) \bullet P(\vec{X}_i/\vec{Z}) + \hat{\vec{X}}_i'(k/k-1) \bullet P(\vec{X}_i/\vec{Z}) \quad (8)$$

where $P(\vec{X}_i/\vec{Z})$ is the probability that given $\vec{Z}(k)$, the i th track did not generate the measurement. If we knew $P(\vec{X}_i/\vec{Z})$, then we could use equation 8 to distribute a single measurement to multiple tracks. Before attempting to derive an expression for $P(\vec{X}_i/\vec{Z})$, let us try to simplify equation 8.

First, either \vec{X}_i did help generate $\vec{Z}(k)$, or it did not. There are no other possibilities. Therefore,

$$P(\vec{X}_i/\vec{Z}) = 1.0 - P(\vec{X}_i/\vec{Z}). \quad (9)$$

Substituting this expression for $P(\vec{X}_i/\vec{Z})$ and the expression for $\hat{\vec{X}}_i'(k/k)$ (equation 7) into equation 8,

$$\begin{aligned} \hat{\vec{X}}_i(k) = & \{ \hat{\vec{X}}_i'(k/k-1) + K_i(k) \bullet [\vec{Z}(k) - H_i \hat{\vec{X}}_i'(k/k-1)] \} \bullet P(\vec{X}_i/\vec{Z}) \\ & + \hat{\vec{X}}_i'(k/k-1) \bullet [1 - P(\vec{X}_i/\vec{Z})] \end{aligned} \quad (10)$$

Simplifying equation 10 and dropping the prime notation on $\hat{\vec{X}}_i'(k/k-1)$ because $\hat{\vec{X}}_i'(k/k-1) = \hat{\vec{X}}_i(k/k-1)$, we get an expression for the composite estimate of the i th target

$$\begin{aligned} \hat{\vec{X}}_i(k) = & \hat{\vec{X}}_i(k/k-1) + K_i(k) \bullet P(\vec{X}_i/\vec{Z}) \\ & \bullet [\vec{Z}(k) - H_i \hat{\vec{X}}_i(k/k-1)]. \end{aligned} \quad (11)$$

This result is identical to the normal estimate correction equation 5 except that the gain is scaled by the probability that \vec{X}_i generated the measurement $\vec{Z}(k)$. If $P(\vec{X}_i/\vec{Z})$ equals one (i.e., we are certain that \vec{X}_i generated $\vec{Z}(k)$), then the modified algorithm reduces to the normal Kalman filter. If, however, we are certain that \vec{X}_i did not generate $\vec{Z}(k)$, then $P(\vec{X}_i/\vec{Z})$ equals zero and the measurement fails to influence the i th track estimate. The effect of $P(\vec{X}_i/\vec{Z})$ is to adjust the bandwidth of the filter based on our certainty that \vec{X}_i generated the measurement. The more uncertain we are, the lower the bandwidth to measurement $\vec{Z}(k)$. This is exactly the behavior we desire for our modified filter. We want the measurement to influence the most likely tracks the most and the least likely tracks the least. All that remains to complete our modification to the kinematic fusion function is to determine how the one measurement/multiple tracks algorithm (PCOMMT) affects the other three filter equations (#2 through #4).

Since we are still propagating the state estimate over the interval (t_{k-1}, t_k) , the process noise errors and propagation errors accounted for in equation 2 still occur. Equation 2 remains unchanged. However, since we have attenuated the gain by $P(\vec{X}_i/\vec{Z})$, we lose some of the effect of the measurement on the state estimate. The error is not reduced as much as it would have been previously. The full error reduction capability of the gain remains unrealized and the error correction equation 4 is incorrect. We must scale the gain, $K_i(k)$, in equation 4 by $P(\vec{X}_i/\vec{Z})$ to reflect the actual correction we are performing in equation 11. The result is that we can achieve the correct modification to equations 4 and 5 by scaling the gain equation 3. The resulting Kinematic Fusion algorithm is

$$\hat{\vec{X}}_i(k/k-1) = \Phi \bullet \hat{\vec{X}}_i(k-1/k-1) + B \bullet \vec{U}(k-1) \quad (12)$$

$$P_i(k/k-1) = \Phi \bullet P_i(k-1/k-1) \bullet \Phi^T + Q_i(k-1) \quad (13)$$

$$K_i(k) = \frac{P(\vec{X}_i/\vec{Z}) \bullet P_i(k/k-1) \bullet H_i^T}{[H_i P_i(k/k-1) H_i^T + R(k)]^{-1}} \quad (14)$$

$$P_i(k/k) = [I - K_i(k) \bullet H_i] \bullet P_i(k/k-1) \quad (15)$$

$$\hat{\vec{X}}_i(k/k) = \hat{\vec{X}}_i(k/k-1) + K_i(k) \bullet [\vec{Z}(k) - H_i \hat{\vec{X}}_i(k/k-1)] \quad (16)$$

Comparing this algorithm with equations 1 through 5, we see that PCOMMT requires only a minor modification to the traditional Kalman filter.

2.2 Attribute Modifications

GE Utica's approach to Attribute Fusion has been to employ the Bayesian recursive algorithm. This technique requires the selection of a complete set of mutually exclusive attributes such as the possibilities of a target being either Friend, Foe, or Neutral. Once we have selected an attribute set, we maintain the probability of each element in the set being correct. Because the set is mutually exclusive,

$$P(\text{Fr}) + P(\text{Fo}) + P(\text{N}) = 1.0 \quad (17)$$

Now let us assume that at time t_{k-1} we know the probability that the i th target is a Foe, $P_i(\text{Fo})_{k-1}$, and we receive a measurement, \vec{Z}_k , containing attribute data, Z_{ak} , at time t_k . How does the traditional Bayesian algorithm estimate the probability that the i th target is a Foe based on this new information? What is $P_i(\text{Fo})_k$? The algorithm employs both Bayes Theorem and the total Probability Theorem to find $P_i(\text{Fo})_k$. Based on a priori knowledge, we can estimate the probability density functions $\rho_i(Z_{ak}|\text{Fo})$, $\rho_i(Z_{ak}|\text{Fr})$, and $\rho_i(Z_{ak}|\text{N})$ for the attribute element Z_{ak} of measurement \vec{Z}_k . Then if we let the a priori probability of target i being a Foe equal $P_i(\text{Fo})_{k-1}$, we can use Bayes Theorem and the Total Probability Theorem (TPT) to write [Ref 2]

$$P_i(\text{Fo})_k = \frac{\rho_i(Z_{ak}|\text{Fo}) \bullet P_i(\text{Fo})_{k-1}}{[\rho_i(Z_{ak}|\text{Fr}) \bullet P_i(\text{Fr})_{k-1} + \rho_i(Z_{ak}|\text{Fo}) \bullet P_i(\text{Fo})_{k-1} + \rho_i(Z_{ak}|\text{N}) \bullet P_i(\text{N})_{k-1}]} \quad (18)$$

where $P_i(\text{Fo})_k$ is really a conditional probability conditioned on the data, but for brevity we have dropped the condition.

If we let

$$\sum_{f=1}^3 \rho_i(Z_{ak}|f) \bullet P_i(f)_{k-1} = \rho_i(Z_{ak}|\text{Fr}) \bullet P_i(\text{Fr})_{k-1} + \rho_i(Z_{ak}|\text{Fo}) \bullet P_i(\text{Fo})_{k-1} + \rho_i(Z_{ak}|\text{N}) \bullet P_i(\text{N})_{k-1} \quad (19)$$

then we can write

$$P_i(\text{Fo})_k = \frac{\rho_i(Z_{ak}|\text{Fo}) \bullet P_i(\text{Fo})_{k-1}}{\sum_{f=1}^3 \rho_i(Z_{ak}|f) \bullet P_i(f)_{k-1}} \quad (20)$$

This is the traditional Attribute Fusion technique: employ the previous probability and the known density functions to calculate the new estimate of $P_i(\text{Fo})_k$. Each element of the attribute set must be updated every time we receive a new measurement.

The traditional method assumes that the i th target generated the measurement, \vec{Z}_k . It assumes perfect correlation. In reality, the update equation should be conditioned on this assumption. That is, given that \vec{X}_i generated the measurement, \vec{Z}_k , the probability of \vec{X}_i being a Foe is

$$P_i(\text{Fo}|\vec{X}_i)_k = \frac{\rho_i(Z_{ak}|\text{Fo}, \vec{X}_i) \bullet P_i(\text{Fo})_{k-1}}{\sum_{f=1}^3 \rho_i(Z_{ak}|f, \vec{X}_i) \bullet P_i(f)_{k-1}} \quad (21)$$

The PCOMMT algorithm does not make the perfect correlation assumption in its Attribute Fusion function so it must correct the traditional Bayesian estimate of $P_i(\text{Fo})_k$. If we use the traditional method to calculate $P_i(\text{Fo}|\vec{X}_i)_k$, then we can correct this estimate for PCOMMT with the TPT and knowledge of $P(\vec{X}_i/\vec{Z})$. The corrected probability is

$$P_i(\text{Fo})_k = P_i(\text{Fo}|\vec{X}_i)_k \bullet P(\vec{X}_i/\vec{Z}) + P_i(\text{Fo}|\vec{X}_i)_k \bullet P(\vec{X}_i/\vec{Z}) \quad (22)$$

Now the probability of the i th track being a Foe, given that it failed to generate a measurement at time t_k , $P_i(\text{Fo}|\vec{X}_i)_k$, has to equal the previous probability of Foe estimate. In the absence of new data we have to keep $P_i(\text{Fo})$ constant,

$$P_i(Fo|\bar{X}_i)_k = P_i(Fo)_{k-1} \quad (23)$$

and as before either \bar{X}_i generated \bar{Z}_k or it did not.

$$P(\bar{X}_i|\bar{Z}) = [1 - P(\bar{X}_i|\bar{Z})] \quad (24)$$

We can write our correction equation 22 as

$$P_i(Fo)_k = P_i(Fo|\bar{X}_i)_k \cdot P(\bar{X}_i|\bar{Z}) + P_i(Fo)_{k-1} \cdot [1 - P(\bar{X}_i|\bar{Z})] \quad (25)$$

As in the kinematic fusion case, we have attenuated our estimate and added in the possibility that perhaps no update should occur because \bar{X}_i may not have generated \bar{Z}_k . The result is a slower, more cautious reaction to the incoming data and the "Low-Pass" analogy applies here as well. The modification, in terms of $P(\bar{X}_i|\bar{Z})$, is given by

$$P_i(Fo)_k = [P_i(Fo|\bar{X}_i)_k - P_i(Fo)_{k-1}] \cdot P(\bar{X}_i|\bar{Z}) + P_i(Fo)_{k-1} \quad (26)$$

Both equations 25 and 26 are equivalent expressions for $P_i(Fo)_k$. They illustrate that, although the modification is not as simple as in the Kinematic Fusion case, it is a minor addition to the traditional Bayesian, attribute fusion approach. Once we have determined $P(\bar{X}_i|\bar{Z})$, the modifications require two more adds and one more multiply per attribute set element. Thus, the PCOMMT algorithm is not only a better method for correlating attribute data to kinematic data, but it also appears to be a feasible solution to the overall correlation problem assuming we can calculate $P(\bar{X}_i|\bar{Z})$.

The exact calculation of $P(\bar{X}_i|\bar{Z})$ is nontrivial. To account for all of the possibilities, computation of $P(\bar{X}_i|\bar{Z})$ is extremely complex.

3.0 DERIVATION OF THE PROBABILITY THAT TRACK_i GENERATED THE MEASUREMENT, $P(\bar{X}_i|\bar{Z})$

To find $P(\bar{X}_i|\bar{Z})$, we must know all of the possible tracks and combinations of tracks that could have generated \bar{Z}_k . We must find the probability of each possibility being correct. Then we can sum all of the possible probabilities that include \bar{X}_i to find $P(\bar{X}_i|\bar{Z})$. The first step is to define all possible ways of generating \bar{Z}_k .

3.1 Possibilities

The first and most likely possibility is that one and only one target in the proximity of the measurement generated \bar{Z}_k . (Note: We consider

false alarms valid targets until we prove otherwise.) The second possibility is that two and only two targets combined to generate the measurement. The third possibility is that three and only three targets combined to generate the measurement. We could go on and on; however, the possibility of more than three targets combining to generate \bar{Z}_k is extremely unlikely, and the inclusion of the possibility into the derivation for $P(\bar{X}_i|\bar{Z})$ significantly increases its complexity. So we assume that if there are more than three targets present, they cannot group themselves so closely that our sensors will fail to discriminate them. In addition, we make the assumption that no new target (a category that includes false alarms as well as valid targets) will combine with other targets to generate the measurement. New targets act alone. In effect, this assumption means that we will ignore the possibility that a new target might combine with an existing target to generate the measurement. If this situation actually occurs, then we will give the entire measurement to the established tracks. With these assumptions, Figure 7 illustrates the possible events. (Note that although the targets themselves generate the measurements, we have shown track positions as estimates of the target state vector because that's all we have to go on.) Figure 7 shows that we ignore targets outside the Association's "Fine Gate", which limits our algorithm to a small number of targets, and we choose to perform PCOMMT on the three to five targets with the lowest Chi-square numbers. It also illustrates the possible combinations of the three targets near the measurement that may have combined to generate \bar{Z}_k . Now we must derive the probabilities that, given \bar{Z}_k , each one of the events pictured could have generated the measurement.

3.2 Probabilities

To illustrate the complex probabilities associated with this problem, we start with the symbolic Venn diagram shown in Figure 8. Now it becomes apparent why we broke the possibilities down into three distinct groups in Figure 7. Figure 8 shows that all of the events occurring within one of the three groups are mutually exclusive. That is, if \bar{X}_i generated the measurement alone, then none of the other events in Group 1 could have occurred. \bar{X}_1 , \bar{X}_3 , \bar{X}_5 , etc., played no part in generating the measurement. Or similarly, given that one and only one target generated \bar{Z}_k , the sum of all the conditional event probabilities within Group 1 must equal one. With this knowledge, we can write all of the possibilities as shown in Table 1.

Table 1 clarifies the probabilities we must derive: The single target probabilities, $P(\bar{X}_i|\bar{Z}.1)$; the double target probabilities, $P(\bar{X}_i \& \bar{X}_j|\bar{Z}.2)$; the new target probability, $P(\bar{X}_{nt}|\bar{Z}.1)$; and the group probabilities, $P(1)$, $P(2)$, and $P(3)$. Some of these derivations are complex, some require sweeping assumptions, and some may result in negligible probabilities. So before we continue, let us discuss the meaning of some of these terms. The most questionable terms in terms of significance are those accounting for the possibility of multiple targets combining to generate a common measurement.

Why account for the possibility of more than one target combining to generate a single measurement? Is this an unlikely possibility? The answer

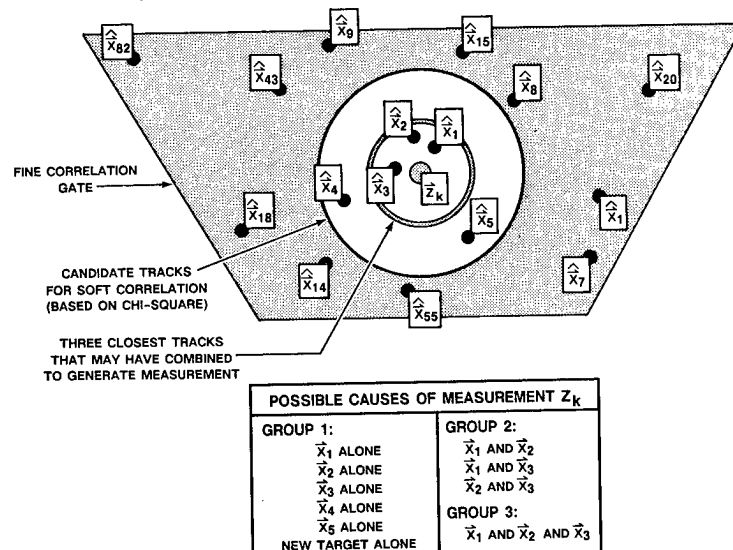
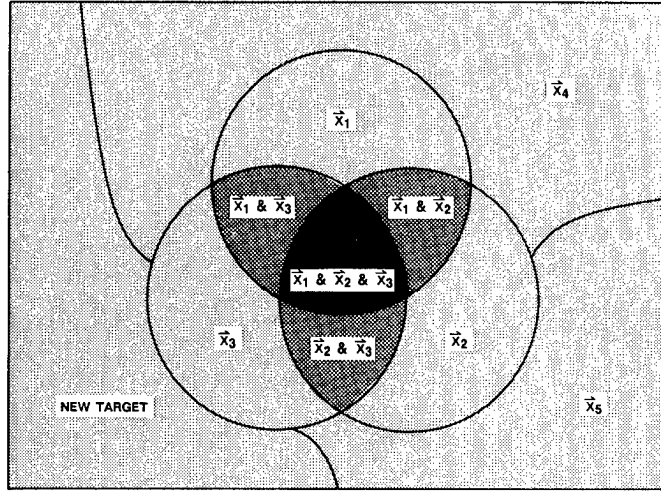


Figure 7. Possible Contributors to Measurement



GROUP 1: ONE AND ONLY ONE
GROUP 2: TWO AND ONLY TWO
GROUP 3: THREE AND ONLY THREE

Figure 8. Venn Diagram of Possibilities

TABLE 1. EVENT PROBABILITIES

	$P(\bar{x}_i \text{ ALONE} \bar{z})$	- GIVEN \bar{z}_k , THE PROBABILITY THAT THE i th TRACK ALONE GENERATED IT.
	$P(\bar{x}_i \bar{z}, 1)$	- GIVEN THAT ONE TARGET GENERATED \bar{z}_k , AND GIVEN THIS PARTICULAR \bar{z}_k , THE PROBABILITY THAT THE GENERATING TRACK WAS THE i th TRACK.
	$P(1)$	- PROBABILITY THAT ONE TARGET ALONE GENERATED \bar{z}_k .
GROUP 1:		
	$P(\bar{x}_1 \text{ ALONE} \bar{z}) = P(\bar{x}_1 \bar{z}, 1) \cdot P(1)$	
	$P(\bar{x}_2 \text{ ALONE} \bar{z}) = P(\bar{x}_2 \bar{z}, 1) \cdot P(1)$	
	$P(\bar{x}_3 \text{ ALONE} \bar{z}) = P(\bar{x}_3 \bar{z}, 1) \cdot P(1)$	
	$P(\bar{x}_4 \text{ ALONE} \bar{z}) = P(\bar{x}_4 \bar{z}, 1) \cdot P(1)$	
	$P(\bar{x}_5 \text{ ALONE} \bar{z}) = P(\bar{x}_5 \bar{z}, 1) \cdot P(1)$	
	$P(\bar{x}_{nt} \text{ ALONE} \bar{z}) = P(\bar{x}_{nt} \bar{z}, 1) \cdot P(1)$	
	$\sum_{i=1}^T P(\bar{x}_i \bar{z}, 1) + P(\bar{x}_{nt} \bar{z}, 1) = 1$	
GROUP 2:		
	$P(\bar{x}_1 \ \& \ \bar{x}_2 \text{ ONLY} \bar{z}) = P(\bar{x}_1 \ \& \ \bar{x}_2 \bar{z}, 2) \cdot P(2)$	
	$P(\bar{x}_1 \ \& \ \bar{x}_3 \text{ ONLY} \bar{z}) = P(\bar{x}_1 \ \& \ \bar{x}_3 \bar{z}, 2) \cdot P(2)$	
	$P(\bar{x}_2 \ \& \ \bar{x}_3 \text{ ONLY} \bar{z}) = P(\bar{x}_2 \ \& \ \bar{x}_3 \bar{z}, 2) \cdot P(2)$	
	$\sum_{i=1}^2 \sum_{j=i+1}^3 P(\bar{x}_i \ \& \ \bar{x}_j \bar{z}, 2) = 1$	
GROUP 3:		
	$P(\bar{x}_1 \ \& \ \bar{x}_2 \ \& \ \bar{x}_3 \bar{z}) = P(\bar{x}_1 \ \& \ \bar{x}_2 \ \& \ \bar{x}_3 \bar{z}, 3) \cdot P(3)$	
	$\frac{P(\bar{x}_1 \ \& \ \bar{x}_2 \ \& \ \bar{x}_3 \bar{z}, 3)}{P(\bar{x}_1 \ \& \ \bar{x}_2 \ \& \ \bar{x}_3 \bar{z}, 3)} = 1$	
	$P(\bar{x}_i \ \& \ \bar{x}_j \text{ ONLY} \bar{z})$	- GIVEN \bar{z}_k , THE PROBABILITY THAT TWO TARGETS, \bar{x}_i AND \bar{x}_j , COMBINED TO GENERATE IT.
	$P(\bar{x}_i \ \& \ \bar{x}_j \bar{z}, 2)$	- GIVEN THAT TWO TARGETS GENERATED \bar{z}_k , AND GIVEN THIS PARTICULAR \bar{z}_k , THE PROBABILITY THAT THE GENERATING TRACKS WERE TRACK i AND TRACK j .
	$P(2)$	- PROBABILITY THAT TWO AND ONLY TWO TARGETS COMBINED TO GENERATE \bar{z}_k .
	$P(\bar{x}_1 \ \& \ \bar{x}_2 \ \& \ \bar{x}_3 \bar{z})$	- GIVEN \bar{z}_k , THE PROBABILITY THAT THE THREE TARGETS COMBINED TO GENERATE THE MEASUREMENT.
	$P(\bar{x}_1 \ \& \ \bar{x}_2 \ \& \ \bar{x}_3 \bar{z}, 3)$	- GIVEN THAT THREE TARGETS GENERATED \bar{z}_k , AND GIVEN THIS PARTICULAR \bar{z}_k , THE PROBABILITY THAT THE GENERATING TRACKS WERE TRACK ₁ AND TRACK ₂ AND TRACK ₃ .
	$P(3)$	- PROBABILITY THAT THREE AND ONLY THREE TARGETS COMBINED TO GENERATE \bar{z}_k .

is yes and no. If the sensor measures unique attribute data (e.g., transmitter frequency), then no two targets will ever combine to generate a common measurement, and the answer is yes, the multiple target probabilities are negligible. If, however, the measurement is not necessarily unique, then the multiple target probabilities are nonnegligible. For example, assume that an infrared sensor has established two tracks very close together in angle (so close that our radar dwells on both targets during a single coherent pulse interval (CPI)). If the targets are very close in range, then the radar will receive only ONE detection. Which track file should get the benefit of this single radar measurement? Since both combined to generate the track, then both should receive the entire \vec{Z}_k . The nearest-neighbor technique would give the entire measurement to only one of the tracks. The PCOMMT algorithm will distribute the measurement across both tracks, but with what weighting function? If we ignore the possibility of more than one target generating a single measurement, then the best value we could achieve for $P(\vec{X}_i | \vec{Z})$ is 0.5 (in other words, give half the measurement to each track). But this is not anywhere near what the tracks should have received. They both combined to generate the radar measurement so they should both receive the entire measurement. $P(\vec{X}_i | \vec{Z})$ should equal 1 for both tracks. Since the possibilities of each target having generated the measurement alone are mutually exclusive, unless we include the possibility of more than one target combining to generate \vec{Z}_k , we can never give the entire measurement to more than one target. Thus, the multiple target probabilities are nonnegligible in this case. They are particularly valuable when dealing with kinematic measurements from a sensor suite containing widely varying degrees of individual sensor accuracy. The derivation for the multiple target probabilities is extremely tedious. We present the derivation for single target probability because it applies directly to the ESM/Radar correlation problem. But we present only a brief derivation for the remaining probabilities in Table 1.

For the ESM/Radar correlation problem, each measurement is unique; it could have only come from one target. All we need to calculate from Table 1 is the Group 1 or single target probabilities. For this application,

$$P(\vec{X}_i | \vec{Z}) = P(\vec{X}_i | \vec{Z}, 1) \quad (27)$$

If we calculate $P(\vec{X}_i | \vec{Z}, 1)$, then we have found $P(\vec{X}_i | \vec{Z})$. Let us derive $P(\vec{X}_i | \vec{Z}, 1)$.

3.3 ESM/Radar Correlation Application Single Target Probabilities - $P(\vec{X}_i | \vec{Z}, 1)$

Although there is a lot of statistical information in Table 1 that must be derived, we already know one very essential piece of information. We know the probability density function (pdf) of the measurement, given the track estimate. Given that the i th track generates the measurement alone, then the distribution of \vec{Z}_k around the estimate $\hat{\vec{X}}_i(k|k-1)$ will be Gaussian with a mean of $H_1 \hat{\vec{X}}_i(k|k-1)$, and a covariance equal to the sum of the estimate's error covariance, $P_1(k|k-1)$, and the measurement noise covariance, R .

$$\rho(\vec{Z} | \vec{X}_i, 1) = \left[(2\pi)^M \cdot |HPH^T + R| \right]^{-0.5} \cdot e^{-0.5 \cdot [Z - H\hat{\vec{X}}]^T \cdot [HPH + R]^{-1} \cdot [Z - H\hat{\vec{X}}]} \quad (28)$$

where the estimate and covariance is valid at the measurement time, t_k , and M is the number of components in \vec{Z} . In terms of the innovation (v_i) and its covariance (S_i), equation 28 becomes

$$\rho(\vec{Z} | \vec{X}_i, 1) = \left[(2\pi)^M \cdot |S_i| \right]^{-0.5} \cdot e^{-0.5 \cdot v_i^T \cdot S_i^{-1} \cdot v_i} \quad (29)$$

or we could write the density function in terms of the Chi-square statistic (χ_i^2) generated in the Association/Correlation function

$$\rho(\vec{Z} | \vec{X}_i, 1) = \left[(2\pi)^M \cdot |S_i| \right]^{-0.5} \cdot e^{-0.5 \cdot \chi_i^2} \quad (30)$$

Equation 30 represents the a priori pdf. Given that \vec{X}_i will generate a measurement, equation 30 is the probability distribution of it generating \vec{Z}_k . We want the a posteriori probability. In other words, given \vec{Z}_k occurred, what is the probability that \vec{X}_i generated it, $P(\vec{X}_i | \vec{Z}, 1)$? Bayes

Theorem and the Total Probability Theorem again allow us to determine an a posteriori from the a priori pdf.

$$P(\vec{X}_i | \vec{Z}, 1) = \frac{\rho(\vec{Z} | \vec{X}_i, 1) \cdot P(\vec{X}_i)}{\sum_{n=1}^T [\rho(\vec{Z} | \vec{X}_n, 1) \cdot P(\vec{X}_n)] + \rho(\vec{Z} | \vec{X}_{nt}, 1) \cdot P(\vec{X}_{nt})} \quad (31)$$

where the $P(\vec{X}_n)$'s are the probabilities that a particular \vec{X}_n will generate a measurement, and T is the number of tracks that qualifies for PCOMMT. The probability that a target will generate a measurement is equal to its probability of detection, which we can approximate from the track's range estimate. We emphasize that $P(\vec{X}_n)$ is the n th track's probability of detection by adding the subscript d . Rewritten, our equation for the probability of the i th track having generated the current measurement, and the cornerstone of the PCOMMT algorithm is

$$P(\vec{X}_i | \vec{Z}, 1) = \frac{\rho(\vec{Z} | \vec{X}_i, 1) \cdot P_d(\vec{X}_i)}{\sum_{n=1}^T [\rho(\vec{Z} | \vec{X}_n, 1) \cdot P_d(\vec{X}_n)] + \rho(\vec{Z} | \vec{X}_{nt}, 1) \cdot P_d(\vec{X}_{nt})} \quad (32)$$

where everything is known in equation 32 except the "new target" terms. We need some reasonable values for them.

3.3.1 New Target Probabilities

Attaching a value to the probability of a new target appearing is guesswork at best. But since an ESM system generally has a much greater range than the radar system it augments, the possibility of new tracks is very real, and we had better make our guesswork as accurate as possible.

Let us take the first term, $P(\vec{Z} | \vec{X}_{nt}, 1)$. This is a Gaussian density described by the form of equation 28. Note that the equation employs the state estimate as the random variable's mean value. But since the target in question is new, we have established no estimate, and in fact if we do initialize an estimate, its first value will equal $\vec{Z}(k)$ so that

$$\vec{Z}(k) - H_1 \hat{\vec{X}}_1(k|k-1) = 0 \quad (33)$$

There are other ways to handle the new target's estimate, but for our case, let us substitute equation 33 into the density function equation 28. Then we have

$$\rho(\vec{Z} | \vec{X}_{nt}, 1) = \left[(2\pi)^M \cdot |R_{nt} + R| \right]^{-0.5} \cdot e^{-0.5 \cdot \vec{Z}^T \cdot (R_{nt} + R)^{-1} \cdot \vec{Z}} \quad (34)$$

where R_{nt} is the new target covariance matrix. Now we can work with R_{nt} to make the pdf small enough to prevent new tracks from stealing measurements from established tracks. If we always initialize our tracks with the measurements, then the estimate's error should become less than the measurement error. The inequality

$$H \cdot P \cdot H^T \leq R \quad (35)$$

should hold for tracks receiving consistent measurements. To be conservative, let us choose

$$R_{nt} = 4 \cdot R \quad (36)$$

to account for setting $H\hat{\vec{X}}(k)$ equal to $\vec{Z}(k)$, and for the possibility of existing tracks having degraded estimates due to sporadic measurements. Then, as a first cut,

$$\rho(\vec{Z} | \vec{X}_{nt}, 1) = \left[(2\pi)^M \cdot |5R| \right]^{-0.5} \quad (37)$$

which is a constant over the region $[(2\pi)^M |5R|]^{-0.5}$, centered on the measurement. This is close to a three-sigma gate for track initialization.

The other unknown term, $P_d(\vec{X}_{nt})$, is difficult to quantify. It is the probability that new targets (including false alarms) will appear. It depends on the environment, the mission, etc. It could conceivably be any value between zero and one, but remember the higher the probability we assign new targets, the lower the existing track probabilities will become. Every application has a different value for $P_d(\vec{X}_{nt})$. It is probably best to start with a low value for new target probability of detection and then quantify $P_d(\vec{X}_{nt})$'s affect on your particular application.

3.4 ESM/Radar Algorithm Summary

This completes the derivation of all the terms required for the ESM/Radar fusion algorithm. Because each ESM measurement is unique and could only originate from a single target, the possibility of more than one target combining to generate $\vec{Z}(k)$ is zero. Therefore, for this case, the Group 2 and Group 3 probabilities listed in Table 1 all equal zero and the probability that one and only one target generated \vec{Z} , $P(1)$, equals one. Thus,

$$P(\vec{X}_i|\vec{Z}) = P(\vec{X}_i|\vec{Z}, 1) \quad (38)$$

and the ESM/Radar fusion process is summarized below:

- *Association/Correlation*
 - Same as traditional "Nearest-Neighbor" algorithm except instead of allowing only the nearest-neighbor to pass, the correlation criterion allows the nearest-neighbors to pass

For EVERY track that passes the Association/Correlation algorithm, the following Kinematic and Attribute Fusion functions must be performed

- *Kinematic Fusion*

$$\begin{aligned} \hat{\vec{X}}_i(k|k-1) &= \Phi \cdot \hat{\vec{X}}_i(k-1|k-1) + B \cdot \vec{U}(k-1) \\ P_i(k|k-1) &= \Phi \cdot P_i(k-1|k-1) \cdot \Phi^T + Q(k-1) \\ K_i(k) &= P(\vec{X}_i|\vec{Z}) \cdot P_i(k|k-1) \cdot H_i^T \\ &\quad \cdot [H_i P_i(k|k-1) H_i^T + R(k)]^{-1} \\ P_i(k|k) &= [I - K_i(k) H_i] \cdot P_i(k|k-1) \\ \hat{\vec{X}}_i(k|k) &= \hat{\vec{X}}_i(k|k-1) + K_i(k) \cdot [\vec{Z}(k) - H_i \cdot \hat{\vec{X}}_i(k|k-1)] \end{aligned} \quad (39)$$

- *Attribute Fusion (Friend/Foe/Neutral Example)*

$$\begin{aligned} P_i(\text{Fo}|\vec{X}_i)_k &= \frac{\rho_i(Z_{ak}|\text{Fo}) \cdot P_i(\text{Fo})_{k-1}}{\sum_{f=1}^3 \rho(Z_{ak}|f) \cdot P_i(f)_{k-1}} \\ P_i(\text{Fr}|\vec{X}_i)_k &= \frac{\rho_i(Z_{ak}|\text{Fr}) \cdot P_i(\text{Fr})_{k-1}}{\sum_{f=1}^3 \rho(Z_{ak}|f) \cdot P_i(f)_{k-1}} \\ P_i(\text{N}|\vec{X}_i)_k &= 1 - P_i(\text{Fo}|\vec{X}_i)_k - P_i(\text{Fr}|\vec{X}_i)_k \end{aligned} \quad (40)$$

Modify:

$$\begin{aligned} P_i(\text{Fo})_k &= P_i(\text{Fo}|\vec{X}_i)_k \cdot P(\vec{X}_i|\vec{Z}) + P_i(\text{Fo})_{k-1} \cdot P(\vec{X}_i|\vec{Z}) \\ P_i(\text{Fr})_k &= P_i(\text{Fr}|\vec{X}_i)_k \cdot P(\vec{X}_i|\vec{Z}) + P_i(\text{Fr})_{k-1} \cdot P(\vec{X}_i|\vec{Z}) \\ P_i(\text{N})_k &= 1 - P_i(\text{Fo})_k - P_i(\text{Fr})_k \end{aligned}$$

where $P(\vec{X}_i|\vec{Z})$ is calculated as shown in Figure 9.

4.0 MULTIPLE TARGET AND RAID CELL ASSESSMENT APPLICATION: MULTIPLE TARGET PROBABILITIES

To find $P(\vec{X}_i|\vec{Z})$ for the situation where measurements are ambiguous and it is possible to assign the entire measurement to a number of tracks, we must calculate the remaining probabilities listed in Table 1. Remember that we have made the assumption that at most, three targets should receive the entire measurement. The following derivation is tedious and requires making a number of assumptions that may or may not apply to the reader's specific situation. We include the high points and results as points of interest.

We begin by looking at the Group 2 probabilities listed in Table 1. We must find the probability of each of the three closest targets combining with one of the other two to generate $\vec{Z}(k)$. (Remember that we have assumed that at most, three targets would combine to generate $\vec{Z}(k)$.) We must find two probabilities: the probability that two targets did generate $\vec{Z}(k)$, $P(2)$; and the probability that given two targets combined to generate $\vec{Z}(k)$ and given $\vec{Z}(k)$, the i th and j th targets were the two, $P(\vec{X}_i \& \vec{X}_j|\vec{Z}, 2)$. Let us start with $P(\vec{X}_i \& \vec{X}_j|\vec{Z}, 2)$.

Given that two targets generated $\vec{Z}(k)$, if we knew the probability distribution of $\vec{Z}(k)$ around the i th and j th target, \vec{X}_i and \vec{X}_j , and if we knew the probability that \vec{X}_i and \vec{X}_j were the two of our three targets that produced a measurement, then we could write the solution for $P(\vec{X}_i \& \vec{X}_j|\vec{Z}, 2)$ as

$$P(\vec{X}_i \& \vec{X}_j|\vec{Z}, 2) = \frac{\rho(\vec{Z}|\vec{X}_i \& \vec{X}_j, 2) \cdot P(\vec{X}_i \& \vec{X}_j \text{ generate } \vec{Z}|2)}{[\rho(\vec{Z}|\vec{X}_1 \& \vec{X}_2, 2) P(\vec{X}_1 \& \vec{X}_2 \text{ generate } \vec{Z}|2) + \rho(\vec{Z}|\vec{X}_2 \& \vec{X}_3, 2) P(\vec{X}_2 \& \vec{X}_3 \text{ generate } \vec{Z}|2) + \rho(\vec{Z}|\vec{X}_1 \& \vec{X}_3, 2) P(\vec{X}_1 \& \vec{X}_3 \text{ generate } \vec{Z}|2)]} \quad (41)$$

where $\rho(\vec{Z}|\vec{X}_i \& \vec{X}_j, 2)$ is the density function that represents the distribution of $\vec{Z}(k)$ about the two tracks that generated it, and $P(\vec{X}_i \& \vec{X}_j \text{ generate } \vec{Z}|2)$ is the probability that the i th and j th targets combine to generate a measurement. Now we have to make some assumptions.

We assume that $\vec{Z}(k)$ is Gaussianly distributed somehow around the two targets that generated it. Let us assume that the distribution's mean is the midpoint between \vec{X}_i and \vec{X}_j as shown in Figure 10. Then the density function we're looking for is

$$\begin{aligned} \rho(\vec{Z}|\vec{X}_i \& \vec{X}_j, 2) &= [(2\pi)^M |S_{dij}|]^{-1/2} \\ &\quad \cdot e^{-\frac{1}{2} \vec{v}_{dij}^T S_{dij}^{-1} \vec{v}_{dij}} \end{aligned} \quad (42)$$

where M is the dimension of $\vec{Z}(k)$ and

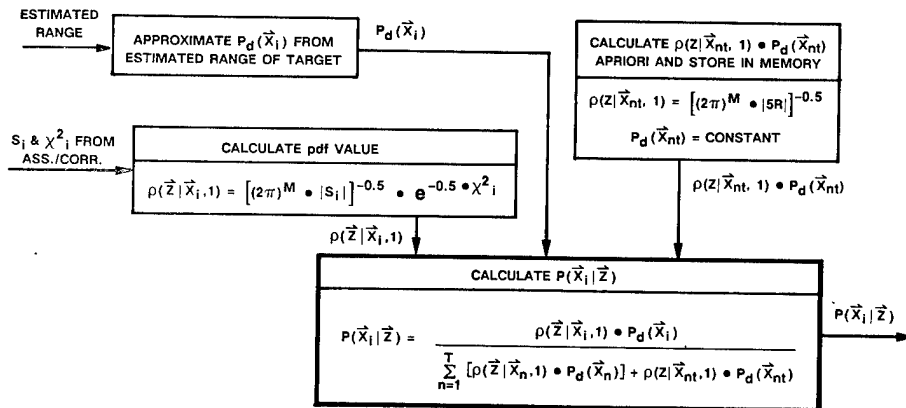


Figure 9. ESM/Radar Probability Calculation

$$\hat{v}_{dij} = \hat{z}(k) - \frac{1}{2} (\hat{H}_i \hat{x}_i + \hat{H}_j \hat{x}_j), \quad (43)$$

and

$$S_{dij} = \frac{1}{4} [H_i P_i H_i^T + H_j P_j H_j^T + H_i P_{ij} H_j^T + H_j P_{ji} H_i^T] + R \quad (44)$$

Note that H_i is the transformation of \hat{x}_i to measurement-space, P_i is the error covariance matrix for \hat{x}_i , $P_{ij} = P_{ji}^T$ is the cross-error covariance matrix between \hat{x}_i and \hat{x}_j , and R is the measurement noise covariance. Next we must find $P(\hat{x}_i \& \hat{x}_j \text{ generate } \hat{z}|2)$.

The probability of two and only two targets generating a measurement depends on the probability of those two targets contributing to any measurement. Assuming at most three targets will contribute to a single measurement gives (from the Venn Diagram, Figure 8)

$$P(\hat{x}_i \& \hat{x}_j \text{ generate } \hat{z}) = P(\hat{x}_i \& \hat{x}_j \text{ will contribute}) - P(3) \quad (45)$$

where $P(\hat{x}_i \& \hat{x}_j \text{ will contribute})$ is the probability that \hat{x}_i and \hat{x}_j contribute to a measurement and $P(3)$ is the probability that all three combine to generate the measurement. We find the conditional probability by conditioning $P(\hat{x}_i \& \hat{x}_j \text{ generate } \hat{z})$ with the event that two and only two targets generate $\hat{z}(k)$.

$$P(\hat{x}_i \& \hat{x}_j \text{ generate } \hat{z}|2) = \frac{P(\hat{x}_i \& \hat{x}_j \text{ will contribute}) - P(3)}{P(2)} \quad (46)$$

$P(1)$ is the probability that one and only one target generated $\hat{z}(k)$ and $P(2)$ is the probability that two and only two generated $\hat{z}(k)$. If we knew $P(\hat{x}_i \& \hat{x}_j \text{ will contribute})$, then we could find $P(1)$, $P(2)$, and $P(3)$. $P(1)$ is the probability that neither two nor three targets combined to generate a measurement.

$$P(1) = \begin{bmatrix} 1 - P(\hat{x}_1 \& \hat{x}_2 \text{ will contribute}) \\ \cdot [1 - P(\hat{x}_2 \& \hat{x}_3 \text{ will contribute})] \\ \cdot [1 - P(\hat{x}_1 \& \hat{x}_3 \text{ will contribute})] \end{bmatrix} \quad (47)$$

From the Venn Diagram

$$P(2) = P(\hat{x}_1 \& \hat{x}_2 \text{ will contribute}) + P(\hat{x}_2 \& \hat{x}_3 \text{ will contribute}) + P(\hat{x}_1 \& \hat{x}_3 \text{ will contribute}) - 3P(3), \quad (48)$$

and because events ①, ②, and ③ make up a mutually exclusive set $P(3) = 1 - P(1) - P(2)$ or

$$P(3) = \frac{1}{2} [P(1) + P(\hat{x}_1 \& \hat{x}_2 \text{ will contribute}) + P(\hat{x}_2 \& \hat{x}_3 \text{ will contribute}) + P(\hat{x}_1 \& \hat{x}_3 \text{ will contribute}) - 1]. \quad (49)$$

We must focus our efforts on finding $P(\hat{x}_i \& \hat{x}_j \text{ will contribute})$.

Let us assume that whenever the distance between targets i and j in the measurement-space is less than some threshold, $\bar{\sigma}$, then they will definitely be ambiguous and they will contribute to a single measurement (if they are within detection range). Then if we define

$$\Delta \hat{x}_{ij} = \hat{H}_i \hat{x}_i - \hat{H}_j \hat{x}_j \text{ and } \Delta \hat{x}_{ij} = \hat{H}_i \hat{x}_i - \hat{H}_j \hat{x}_j.$$

we have

$$P(\hat{x}_i \& \hat{x}_j \text{ will contribute}) = P(-\bar{\sigma} < \Delta \hat{x}_{ij} < \bar{\sigma} | \det) \cdot P_d(\hat{x}_i \& \hat{x}_j) \quad (50)$$

where $P(-\bar{\sigma} < \Delta \hat{x}_{ij} < \bar{\sigma} | \det)$ is the probability that given $\hat{x}_i \& \hat{x}_j$ are detectable, they combine and generate a single measurement because $\Delta \hat{x}_{ij}$ is less than $\bar{\sigma}$, and $P_d(\hat{x}_i \& \hat{x}_j)$ is the probability of detecting the combined $\hat{x}_i \& \hat{x}_j$ target. We reason that the actual value for $\Delta \hat{x}_{ij}$ is Gaussianly distributed about $\Delta \hat{x}_{ij}$, then

$$P_{\Delta \hat{x}_{ij}}(\Delta \hat{x}_{ij}) = \left[(2\pi)^M |S_{\Delta \hat{x}_{ij}}| \right]^{-1/2} \cdot e^{-\frac{1}{2} (\Delta \hat{x}_{ij} - \Delta \hat{x}_{ij})^T S_{\Delta \hat{x}_{ij}}^{-1} (\Delta \hat{x}_{ij} - \Delta \hat{x}_{ij})} \quad (51)$$

where

$$P(\hat{x}_i | \hat{z}) = P(\hat{x}_i | \hat{z}, 1) \cdot P(1) + [P(\hat{x}_i \& \hat{x}_2 | \hat{z}, 2) + P(\hat{x}_i \& \hat{x}_3 | \hat{z}, 2)] \cdot P(2) + P(3)$$

GROUP 1 PROBABILITY

$$P(\hat{x}_i | \hat{z}, 1) = \frac{\rho(\hat{z} | \hat{x}_i, 1) \cdot P_d(\hat{x}_i)}{\sum_{n=1}^T [\rho(\hat{z} | \hat{x}_n, 1) \cdot P_d(\hat{x}_n)] + \rho(\hat{z} | \hat{x}_{nt}, 1) \cdot P_d(\hat{x}_{nt})}$$

WHERE,

$$\rho(\hat{z} | \hat{x}_n, 1) = \left[(2\pi)^M |S_{ij}| \right]^{-0.5} e^{-0.5 \cdot X_i^2}$$

$$\rho(\hat{z} | \hat{x}_{nt}, 1) = \left[(2\pi)^M \cdot |SR| \right]^{-0.5}$$

$$P_d(\hat{x}_{nt}) = \text{CONSTANT}$$

$$P_d(\hat{x}_i) = \text{ESTIMATE}$$

GROUP 2 PROBABILITY

$$P(\hat{x}_i \& \hat{x}_j | \hat{z}, 2) = \frac{\rho(\hat{z} | \hat{x}_i \& \hat{x}_j, 2) \cdot P(\hat{x}_i \& \hat{x}_j \text{ generate } \hat{z} | 2)}{[\rho(\hat{z} | \hat{x}_1 \& \hat{x}_2, 2) P(\hat{x}_1 \& \hat{x}_2 \text{ generate } \hat{z} | 2) + \rho(\hat{z} | \hat{x}_2 \& \hat{x}_3, 2) P(\hat{x}_2 \& \hat{x}_3 \text{ generate } \hat{z} | 2) + \rho(\hat{z} | \hat{x}_1 \& \hat{x}_3, 2) P(\hat{x}_1 \& \hat{x}_3 \text{ generate } \hat{z} | 2)]}$$

WHERE

$$\rho(\hat{z} | \hat{x}_i \& \hat{x}_j, 2) = \left[(2\pi)^M |S_{dij}| \right]^{-0.5} \cdot e^{-0.5 \cdot \hat{v}_{dij}^T S_{dij}^{-1} \hat{v}_{dij}}$$

$$P(\hat{x}_i \& \hat{x}_j \text{ generate } \hat{z} | 2) = \frac{P(\hat{x}_i \& \hat{x}_j \text{ will contribute}) - P(3)}{P(2)}$$

$$P(\hat{x}_i \& \hat{x}_j \text{ will contribute}) = P(-\bar{\sigma} < \Delta \hat{x}_{ij} < \bar{\sigma} | \det) \cdot P_d(\hat{x}_i \& \hat{x}_j)$$

$$P(-\bar{\sigma} < \Delta \hat{x}_{ij} < \bar{\sigma} | \det) = \int_{-\bar{\sigma}}^{\bar{\sigma}} \int_{-\bar{\sigma}}^{\bar{\sigma}} \int_{-\bar{\sigma}}^{\bar{\sigma}} \rho_{\Delta \hat{x}_{ij}}(\Delta \hat{x}_{ij}) d(\Delta x_1) d(\Delta x_2) \dots d(\Delta x_M)$$

$$\rho_{\Delta \hat{x}_{ij}}(\Delta \hat{x}_{ij}) = \left[(2\pi)^M |S_{\Delta \hat{x}_{ij}}| \right]^{-0.5} \cdot e^{-0.5 (\Delta \hat{x}_{ij} - \Delta \hat{x}_{ij})^T S_{\Delta \hat{x}_{ij}}^{-1} (\Delta \hat{x}_{ij} - \Delta \hat{x}_{ij})}$$

GROUP PROBABILITIES

$$P(1) = [1 - P(\hat{x}_1 \& \hat{x}_2 \text{ will contribute})] \cdot [1 - P(\hat{x}_2 \& \hat{x}_3 \text{ will contribute})] \cdot [1 - P(\hat{x}_1 \& \hat{x}_3 \text{ will contribute})]$$

$$P(2) = P(\hat{x}_1 \& \hat{x}_2 \text{ will contribute}) + P(\hat{x}_2 \& \hat{x}_3 \text{ will contribute}) + P(\hat{x}_1 \& \hat{x}_3 \text{ will contribute}) - 3P(3)$$

$$P(3) = \frac{1}{2} [P(1) + P(\hat{x}_1 \& \hat{x}_2 \text{ will contribute}) + P(\hat{x}_2 \& \hat{x}_3 \text{ will contribute}) + P(\hat{x}_1 \& \hat{x}_3 \text{ will contribute}) - 1]$$

Figure 10. $\hat{z}(k)$ Location Assumption

$$S_{\Delta X_{ij}} = H_i P_i H_i^T + H_j P_j H_j^T - H_i P_{ij} H_j^T - H_j P_{ji} H_i^T. \quad (52)$$

$P_d(\vec{X}_i \& \vec{X}_j)$ we will have to estimate based on target range estimates, and $P(-\vec{\sigma} < \Delta \vec{X}_{ij} < \vec{\sigma} | \det)$ is the integral of the density function

$$P(-\vec{\sigma} < \Delta \vec{X}_{ij} < \vec{\sigma} | \det) = \int_{-\sigma_M}^{\sigma_M} \int_{-\sigma_2}^{\sigma_2} \int_{-\sigma_1}^{\sigma_1} \rho_{\Delta \vec{X}_{ij}}(\Delta \vec{X}_{ij}) d(\Delta X_1) d(\Delta X_2) \dots d(\Delta X_M) \quad (53)$$

which would require a multidimensional numerical integration. If the off-diagonal terms in $S_{\Delta \vec{X}_{ij}}$ are relatively small then we could say that the

components $\Delta X_1, \Delta X_2, \dots, \Delta X_M$ are uncorrelated and write

$$P(-\vec{\sigma} < \Delta \vec{X}_{ij} < \vec{\sigma} | \det) = \int_{-\sigma_M}^{\sigma_M} \rho_{\Delta X_M}(\Delta X_M) d(\Delta X_M) \dots \int_{-\sigma_2}^{\sigma_2} \rho_{\Delta X_2}(\Delta X_2) d(\Delta X_2) \int_{-\sigma_1}^{\sigma_1} \rho_{\Delta X_1}(\Delta X_1) d(\Delta X_1) \quad (54)$$

where

$$\rho_{\Delta \vec{X}_n}(\Delta X_n) = \frac{1}{\sqrt{2\pi S_{\Delta X_{nn}}}} e^{-(\Delta X_n - \Delta \hat{X}_n)^2 / (S_{\Delta X_{nn}})} \quad (55)$$

4.1 Ambiguous Track/Raid Cell Algorithm Summary

This completes the derivation of the terms we require to compute $P(\vec{X}_i / \vec{Z})$ in the ambiguous target case. In summary, the modifications we must make to the Association/Correlation, the Kinematic Fusion, and the Attribute Fusion functions are the same as the ones we make in the ESM/Radar problem. The difference is that the calculation of $P(\vec{X}_i / \vec{Z})$

is much more tedious. The probability $P(\vec{X}_i / \vec{Z})$ when \vec{X}_i is one of the three closest targets to $\vec{Z}(k)$ is (for discussion let $i=1$)

$$P(\vec{X}_1 / \vec{Z}) = P(\vec{X}_1 | \vec{Z}, 1) \cdot P(1) + [P(\vec{X}_1 \& \vec{X}_2 | \vec{Z}, 2) + P(\vec{X}_1 \& \vec{X}_3 | \vec{Z}, 2)] \cdot P(2) + P(3) \quad (56)$$

where we have summarized the necessary calculations in Figure 11. The calculations are extensive and their complexity probably outweighs the benefits which probabilistic data association brings to the Multiple Target/Raid Cell Assessment problem.

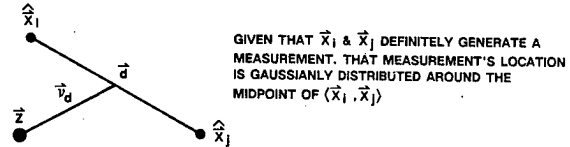


Figure 11. Ambiguous Target Solution

5.0 CONCLUSIONS

When applied to situations where each measurement is uniquely associated to a single source, such as the ESM/Radar problem, probabilistic data association offers the cautious approach required for fusing ID data to kinematic tracks. But when the sources may be ambiguous and the measurement is not unique, the effort required to calculate $P(\vec{X}_i / \vec{Z})$ is too extensive to perform in real-time. Thus, each application requires a thorough analysis before applying probabilistic data association techniques.

6.0 REFERENCES

- [1] Bar-Shalom, Yarkov. *Multitarget - Multisensor Tracking*. Box U-157, Storrs, CN. YBS, 1986.
- [2] Papoulis, Athanasios. *Probability, Random Variables, and Stochastic Processes*, McGraw-Hill, New York, 1965.

MULTIPLE SENSOR TRACKING OF CLUSTERS AND EXTENDED OBJECTS

O. E. Drummond, S. S. Blackman, K. C. Hell

Hughes Aircraft Company
2141 E. Rosecrans Blvd. (Mail Station E52/C227)
El Segundo, CA 90245
Phone (213) 616-2624

This paper develops a group tracking algorithm for the tracking of clusters of closely spaced targets as viewed by passive sensors. This problem is complex because the size and shape of the observed cluster will differ from sensor to sensor. Since a passive sensor provides a projection of the objects, passive sensors in different locations will provide different projections of the 3-dimensional (3-D) shape of a cluster. Furthermore, the number of resolved objects in a cluster can vary as closely spaced targets may appear to a sensor as a single extended object.

In order to track the target clusters with multiple sensors, a method is needed to characterize the size, shape and location of each cluster. The method described in this paper models the target cluster in 3-D as an ellipsoid with the projection in 2-D being an ellipse. The centroid and parameters of the ellipsoid are tracked over time using measurements from widely spaced passive sensors. The filtering methods are described and performance is presented based upon tracking simulations. Application of this method to tracking extended objects with multiple sensors is also discussed.

MULTIPLE SENSOR TRACKING OF CLUSTERS AND EXTENDED OBJECTS

O. E. Drummond, S. S. Blackman, K. C. Hell

Hughes Aircraft Company
2141 E. Rosecrans Blvd. (Mail Station E52/C227)
El Segundo, CA 90245
Phone (213) 616-2624

INTRODUCTION

This paper presents a group tracking method that has been developed for surveillance, such as for the midcourse stage of the Strategic Defense Initiative (SDI). A cluster of targets, consisting of one reentry vehicle (RV) and 40 decoys, is assumed to be deployed from a post boost vehicle (bus). Initially, the members of the cluster will have small random velocities with respect to the center of mass of the cluster. However, as the cluster separates the members of the cluster will experience different gravitational forces due to their different positions within the earth's gravitational force field. This phenomenon, denoted the gravity gradient effect, will tend to further increase the relative velocities and to spread the cluster further apart along the direction of the gravitational force.

As addressed in this paper the tracking process will be performed using observations from passive Electro Optical (EO) sensors. The EO sensors measure the angles from the platform to the target. Then, assuming the platforms are typically separated by large distances, it is possible to determine the three components of a target's position by the process of triangulation (stereo tracking). However, as discussed below, it will not be possible to initially track the members of the closely spaced target cluster as individuals.

In general, it is both inaccurate and wasteful of computer resources to attempt to form individual tracks on closely spaced targets that have essentially the same velocity [1]. This is particularly true if, as in this case, the individual targets may not be resolvable. Note that for EO sensors there are only two dimensions for resolution and that, since different sensor systems will have different viewing angles, the particular set of targets that is resolvable will vary with the sensor. Thus, for the purposes of track stability and efficient resource allocation, group tracking is desirable.

Due to sensor measurement resolution limits, closely spaced targets may appear to the sensor as a single extended object. Typical resolution history is illustrated by Figure 1, which shows the number of resolved targets plus clumps observed by a sensor as a function of time from deployment. A clump is defined to be an extended object containing at least two unresolved targets. The total number of resolved targets plus clumps is the number of elements in a measurement of a group (or cluster) of targets. Results given in Figure 1 are for a cluster of 40 targets and are shown for two spherically uniform distributions of the random velocity with respect to the centroid. The limits on the magnitude of the velocity are 0.5 and 2.0 meters/sec for the upper curve and 0 and 2.0 meters/sec for the lower curve.

In general, it may be possible to estimate the number of individual targets contained in an extended object based upon the shape and/or brightness of the extended object. However, for the purpose of this study, we considered the worst case where an extended object was indistinguishable from an individual target. Thus, the problem is to develop a tracking method for a cluster (or group) of objects in which some closely spaced elements may not be resolvable by the sensors at certain viewing angles. The term cluster will be used to refer to a collection of objects with similar state vectors. The term group will be used to refer to an apparent cluster of objects based on sensor measurements.

The open literature was searched for a group tracking approach based on multiple, passive sensors. Of particular interest is an approach that is applicable to birth-to-death tracking and central-level processing [3]. While there are a number of group tracking approaches, none appear to be suitable for this task.

In developing the new approach, a number of critical issues had to be addressed, as follows. How to represent

the cluster in 3-dimensions (3-D) and in 2-dimensions (2-D) so that the 3-D parameters can be estimated from 2-D parameters? How to track the 3-D parameters over time and how to compute the 2-D parameters from the passive sensor measurements? And finally, how to accommodate the sensor-to-sensor differences in the apparent size, shape and composition of the group and the time varying characteristics of the cluster? The following sections describe the concept that was developed and results of the feasibility simulation that was conducted.

THE GROUP TRACKING APPROACH

The group tracking concept that has been developed characterizes the target cluster by its centroid and extent about the centroid. The extent is represented in three dimensions by an ellipsoid which becomes an ellipse when viewed in the two dimensional focal plane of an EO sensor (Figure 2). As a consequence, it was possible to develop a method to track the parameters of the ellipsoid using multiple separated EO sensors.

The group centroid can be tracked with respect to the Cartesian Earth Centered Inertial (ECI) coordinate system. This estimated centroid defines the center of the ellipsoid enclosing the group. The extent of the ellipsoid is defined by the displacement vectors

$$\delta \underline{x}^T = [\delta x, \delta y, \delta z]$$

$\delta x, \delta y, \delta z$ = displacements from ellipsoid center such that

$$\delta \underline{x}^T P_{xyz}^{-1} \delta \underline{x} \leq D \quad (1a)$$

where

$$P_{xyz} = \begin{bmatrix} \sigma_x^2 & \sigma_{xy}^2 & \sigma_{xz}^2 \\ \sigma_{xy}^2 & \sigma_y^2 & \sigma_{yz}^2 \\ \sigma_{xz}^2 & \sigma_{yz}^2 & \sigma_z^2 \end{bmatrix} \quad (1b)$$

$$\sigma_x^2 = E[\delta x^2], \sigma_{xy}^2 = E[\delta x \cdot \delta y], \text{ etc.}$$

Complex, but readily obtained, expressions can be used to relate the angles (η, ϵ) measured by the EO sensors to the target position (x, y, z) in the ECI coordinate system.

$$\eta = f_1(x, y, z), \epsilon = f_2(x, y, z)$$

Thus, we can define

$$H = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \end{bmatrix} \quad (2)$$

where

$$h_{11} = \frac{\partial f_1}{\partial x}, h_{12} = \frac{\partial f_1}{\partial y}, h_{21} = \frac{\partial f_2}{\partial x}, \dots$$

Then, the angular displacements from the centroid are related to the ECI coordinate displacements through

$$\begin{pmatrix} \delta h \\ \delta \epsilon \end{pmatrix} = H \begin{pmatrix} \delta x \\ \delta y \\ \delta z \end{pmatrix}$$

so that

$$P_{\eta\epsilon} = \begin{bmatrix} \sigma_\eta^2 & \sigma_{\eta\epsilon}^2 \\ \sigma_{\eta\epsilon}^2 & \sigma_\epsilon^2 \end{bmatrix} = H P_{xyz} H^T \quad (3)$$

where

$$\sigma_\eta^2 = E[\delta \eta^2], \sigma_{\eta\epsilon}^2 = E[\delta \eta \cdot \delta \epsilon]$$

The approach is to form a group measurement each time a new set of observations is received. First, in order to form a measurement gate about the predicted centroid, an estimate, $\hat{P}_{\eta\epsilon}$, of the ellipse parameters can be obtained using Eq. 3 and the estimated ellipsoid parameters. Then, the measurement distribution is defined by the ellipse parameters and the sensor measurement error variances. Assuming the contribution from the measurement error to be small relative to the dispersion, a measurement with angular displacements $(\delta \eta, \delta \epsilon)$ from the predicted centroid will satisfy the gate if

$$(\delta \eta, \delta \epsilon) \hat{P}_{\eta\epsilon}^{-1} \begin{pmatrix} \delta \eta \\ \delta \epsilon \end{pmatrix} \leq G \quad (4)$$

The gating constant, G , in Eq. 4 was chosen to be 10 for the results to be presented below.

Thus, to summarize, all observations satisfying Eq. 4 will be considered for use in updating the group track in question. In general, an observation may satisfy multiple group track gates so that a conflict resolution logic is required. However, for this preliminary study we have only considered a single group track. Finally, all observations that are assigned to the group track, by virtue of satisfying Eq. 4 for this case, are then used to form a group observation consisting of a measured centroid and dispersion ellipse. The measured centroid and dispersion parameters are defined in the standard manner, for N individual observations included in the group observation,

$$\bar{\eta}_0 = \frac{1}{N} \sum_{i=1}^N \eta_i, \sigma_{\eta 0}^2 = \frac{1}{N} \sum_{i=1}^N (\eta_i - \bar{\eta}_0)^2 \quad (5)$$

with similar expressions for

$$\bar{\epsilon}_0, \sigma_{\epsilon 0}^2, \sigma_{\eta \epsilon 0}^2$$

The measured quantities are used to update the group track estimates using the Kalman filtering techniques described in the next section. The entire process is summarized by the flow chart shown in Figure 3.

KALMAN FILTERING METHODS

Separate Kalman filters can be used for estimating the target centroid and ellipsoid extent parameters. Both filters, outlined below, use standard extended Kalman filtering methods.

Centroid Filter

The centroid filter is a six state filter using position and velocity for the three Cartesian components in the ECI coordinate system. Extrapolation was done in ten second intervals. Extrapolation over this interval is performed using a trajectory in which the gravitational force at the center of the extrapolation period is computed and then used as a constant throughout the interval. Results obtained using this approximate extrapolation method have been found to closely correspond to those derived using more exact methods.

The initial position and velocity estimates are taken to be the same as those for the bus which deploys the target cluster. It is assumed that the bus is being tracked and the initial covariance matrix elements are chosen using the expected bus tracking accuracies.

The elements of the measurement vector are the centroid azimuth and elevation angles as computed according to Eq. 5. The measurement vector is related to the state estimation vector in the standard manner.

$$y = Hx + v \quad (6)$$

where H is defined in Eq. 2, x is the state vector and v is measurement noise. The measurement noise covariance matrix is taken to be a function of the actual dispersion and the number of elements in the group measurement. The centroid measurement noise is assumed to decrease as more elements of the group are detected. The process noise covariance matrix, Q, was taken to have the form associated with the discretized, continuous-time white process noise [2] and with the magnitude of the acceleration variance chosen to compensate for the effects of the gravity gradient.

Extent Parameter Filter

The extent parameter filter also uses six states. These are the parameters given in Eq. 1 that define the ellipsoid. Using Eq. 3 it is straightforward to find a linear relationship between the measurement vector y and the state estimation vector x

$$y = H_m x + v \quad (7)$$

where

$$y^T = [\sigma_{\eta 0}^2, \sigma_{\eta \epsilon 0}^2, \sigma_{\epsilon 0}^2]$$

$$x^T = [\sigma_x^2, \sigma_{xy}^2, \sigma_{xz}^2, \sigma_y^2, \sigma_{yz}^2, \sigma_z^2]$$

and H_m is a three by six matrix whose elements are found using Eq. 3. Again, v represents the associated measurement noise.

The elements of the target cluster were deployed from the bus with random velocities relative to the velocity of the centroid. Therefore, initially, before the effects of the gravity gradient become appreciable, the cluster extent grows linearly with time. This means that the extent parameters grow with the square of time since deployment. For this study it was assumed that an accurate estimate of the time of deployment could be obtained. Thus, the transition matrix for the process from time step k to $k+1$ was taken to be

$$\Phi(k+1, k) = a^2 I$$

where I is the six by six identity matrix and

$$a = \frac{t_{k+1}}{t_k}$$

As the cluster spreads apart, the effect of the gravity gradient is to further increase the dispersion rate along the direction of the gravitational force. An approximate model was derived and used to aid the filter through increasing the process noise covariance matrix to account for the perturbations due to the gravity gradient. A more exact method in which the actual dispersion rate is estimated taking into account the known effect of the gravity gradient may improve performance. However, as will be illustrated by results to follow, the artifice of making the process noise covariance matrix a function of the gravity gradient leads to an adequate response of the filter to this effect.

As discussed in [1] the measurement noise covariance matrix was made a function of the expected dispersion and the number of elements in the group measurement. As with the centroid measurement, the extent parameter measurement accuracy is assumed to increase with the number of elements in the group measurement. Finally, the initial state estimates and covariance matrix were defined based upon expected random deployment velocity statistics.

THE EXTENDED OBJECT TRACKING APPROACH

An extended object can also be tracked with the multiple passive sensors using this same approach. There are a number of methods that can be used to establish the parameters of an ellipse and the centroid for an extended object as observed by a single passive sensor. The extended object can be represented by an ellipsoid in 3-D just as was described for a cluster.

Using multiple sensors, Eq. 7 can be used to estimate the parameters of the ellipsoid. The centroid in 3-D can be estimated using Eq. 6. Thus, Kalman filters can be used to track the 3-D parameters. However, if the actual size in 3-D of an object is varying, then the dynamic equation used in the Kalman filter must account for that change.

SIMULATION STUDY

A simulation was developed to show feasibility of the group tracking method defined above. This simulation considered a typical 30 min (1800 sec) scenario with a single, isolated cluster as described below.

Scenario Description

For this study, two sensor platforms were observing the dispersion of 40 target objects which were deployed from a target bus at an altitude of 500 km. It was assumed that the bus was previously tracked as an individual target and that group tracking of the 40 objects was initiated at this altitude. The target objects were each given an initial relative velocity to simulate ejection. The relative velocities were imparted in a random direction with magnitudes uniformly distributed between 0 and 2 m/s. The target objects were on a ballistic trajectory. The sensor platforms were in circular orbits with altitudes of 3000 km. One orbit was inclined 40°, the other 80°.

The angular resolution of the sensors was taken to be 0.001 degrees in azimuth and elevation. However, to simplify this preliminary study no measurement errors were introduced.

Coordinate System Definition

The position of the target in the sensor field-of-view (FOV) was defined by two angles, azimuth and elevation. These angles are measured in the Spacecraft Center True of Date (SCT) coordinate system which is centered at the current spacecraft location. The fundamental plane of the SCT system is the plane of the orbit and the azimuthal angle points toward nadir. Azimuth and elevation of the target are then defined in this coordinate system as shown in Figure 4.

Presentation of Results

Results will be presented for the case where the two sensors observe the target cluster every 100 sec. Results derived for shorter (10,50 sec) sampling intervals were somewhat better but the performance improvement did not seem to compensate for the increased computational requirements needed to process the additional data.

Two resolution conditions were examined. For the first (infinite resolution) it was assumed that all targets were observed each look. For the second case (finite resolution) all targets within 0.001 degrees were assumed to be unresolvable. This led to the formation of extended objects which were clumps of unresolved closely spaced targets. For this study extended objects were used by the measurement process as though they were indistinguishable from single targets. A typical resolution history for this case is shown by the lower curve in Figure 1.

Results are shown for typical runs. No Monte Carlo statistics were compiled. However, the same type of results have been obtained for a number of individual runs in which the random number sequences have been

varied. The first results, given in Figure 5, show the total distance from the centroid estimate to the true target group centroid. For the case of infinite resolution, the centroid estimation error is essentially zero but when resolution effects are considered the distance is in the order of 100 to 200 meters throughout the run.

We next consider performance of the extent estimation process. First, Figure 6 shows the ratio of the estimated to the true volume of the ellipsoid defining the target group extent. The volume of the ellipsoid defined by Eq. 1 is given by

$$v = \frac{4\pi}{3} \sqrt{|P_{xyz}|} D^{3/2}$$

Thus, the volume ratio is defined as

$$R_v = \frac{\Delta}{\left[\frac{\hat{P}_{xyz}}{|P_{xyz}|} \right]^{1/2}}$$

where P_{xyz} is computed using the true target positions and \hat{P}_{xyz} is computed using the state vector estimates.

Referring to Figure 6, note that R_v is close to unity throughout the run when all targets are resolved. However, this ratio is considerably larger than unity during the early stages of the run when finite resolution is considered. This is because the objects with smaller dispersion velocities tend to be those that remain closely spaced and are thus seen as extended objects rather than individuals. This tends to distort the apparent ellipsoid by putting less weight on those group members that are close to the centroid. Thus, the apparent ellipsoid dispersion parameters are computed to be too large. This problem could be reduced if extended objects were recognized as consisting of more than one object and weighted accordingly.

In addition to considering the total volume it is important to compare the shape and orientation of the estimated with respect to the true ellipsoid. This comparison was performed by computing the eigenvalues of the estimated dispersion matrix relative to the true dispersion matrix. This is achieved by solving

$$|\hat{P}_{xyz} - \lambda P_{xyz}| = 0$$

for all three solutions for the eigenvalue, λ . This is equivalent to computing the length along the principle axes of the estimated ellipsoid given that the true ellipsoid is a sphere. Thus the eigenvalues can be viewed as the extreme dispersion ratios and the ideal values are unity. Figures 7 and 8 show these results. Again, the accuracy of the estimation process when all targets are resolved is excellent but the distortions associated with the finite resolution case are apparent.

Probably the most important test of the estimation process is the success with which the estimates of the ellipsoid and the projected ellipses actually contain the target group. First, the theoretical probability of a true group member falling outside the ellipsoid defined by Eq. 1 can be computed as a function of D using chi-square statistics with three degrees of freedom. For example, choosing $D = 8.0$ and 13.0 gives probabilities of 0.05 and 0.005 , respectively, of a true target position falling outside the volume defined by Eq. 1. Results indicated that throughout the duration of the run and for both finite and infinite resolution there were essentially no targets outside the volume defined by $D=13$. Also, an average of about 1.5 targets per 40 target cluster were outside the volume defined by $D=8.0$. These results agree well with the theoretical predictions.

The gate defined by Eq. 4 with gating constant $G=10$ would be expected to contain a true target return with probability of about 0.995 . This is computed from chi-square variable statistics with two degrees of freedom. In practice, the simulation results showed that there were no returns outside the gate for either the finite or the infinite resolution cases. This was true for each of the 18 observations (received at 100 sec intervals over the 1800 sec tracking period).

Figures 9 through 12 illustrate gating behavior by showing the gates and the targets positions as projected onto the field of view of one sensor. Results are shown at 400 and 1600 sec. and for the finite and infinite resolution cases. It can be noted that the shape of the cluster changes with time due to the effects of the gravity gradient but that the shape of the gate also changes accordingly. Note the dispersion is reflected in the change of scale between Figures 9 and 10 and Figures 11 and 12.

Referring to Figures 10 and 12, the symbols Δ and \square refer to extended objects, as formed by the finite resolution case, consisting of two and more than two targets, respectively. The effect of unresolved targets is seen to be an increase in the gate size, but for all cases the gates correspond well to the true target positions.

The results presented above used a uniform initial velocity distribution with maximum relative velocity of 2m/s . Other runs were made in which the maximum velocity was increased to 5m/s and the method performed equally well.

CONCLUSION/FUTURE STUDY

This paper has summarized the preliminary development and evaluation of a new group tracking algorithm for use in distributed, passive sensor applications. It has demonstrated the feasibility of this approach for group tracking. In particular, a close match between the estimated ellipsoid taken to represent the target cluster and the true target positions has been found. This conclusion

is based upon the volume and shape of the estimated ellipsoid and the high proportion of targets enclosed by the estimated ellipsoid. Also, gating tests based upon the elliptical projection of the ellipsoid onto the sensor field of view have been shown to include the true target returns.

Much future study is required to further define the method. Logic for the transition from group to individual target tracking have yet to be tested. Also, the concepts that have been identified to handle crossing groups and merging groups must be developed and tested. Corrections to the measurement vector must be added to adjust for the bias caused by measurement errors. Finally, it may be desirable to more accurately represent the effects of the gravity gradient that becomes an important factor in group dispersion as the targets separate.

ACKNOWLEDGEMENT

Many colleagues at Hughes Aircraft, too numerous to list, have contributed to this work. This study was funded by IR&D and other internal Hughes funds.

REFERENCES

1. S. S. Blackman, Chapt. 11, Multiple Target Tracking with Radar Applications, Artech House, Dedham MA (1986).
2. Y. Bar-Shalom and T. E. Fortmann, Section 2.8, Tracking and Data Association, Academic Press, San Diego (1987).
3. O.E. Drummond and S.S. Blackman, "Multiple Sensor, Multiple Target Tracking Challenges of the Strategic Defense Initiative", 1st National Symposium on Sensor Fusion, Orlando, FL, April 8, 1988.

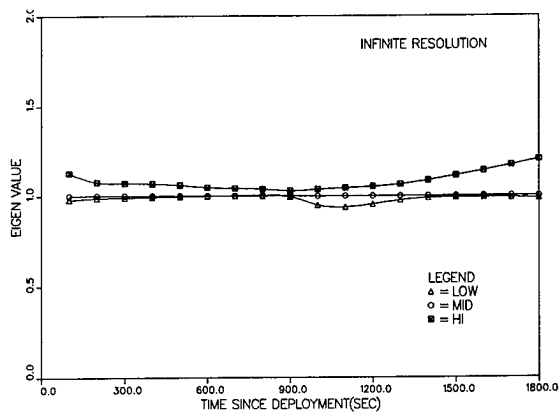


Figure 7. Infinite Resolution Dispersion Estimate Eigenvectors

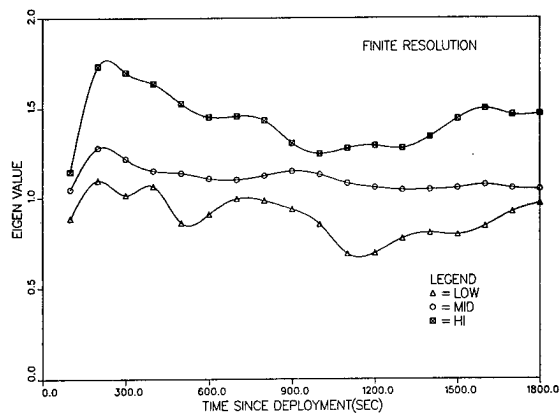


Figure 8. Finite Resolution Dispersion Estimate Eigenvectors

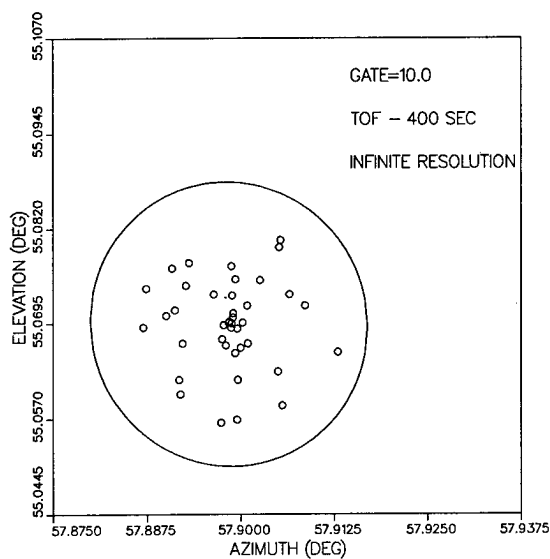


Figure 9. Gating at 400 sec for Infinite Resolution Case

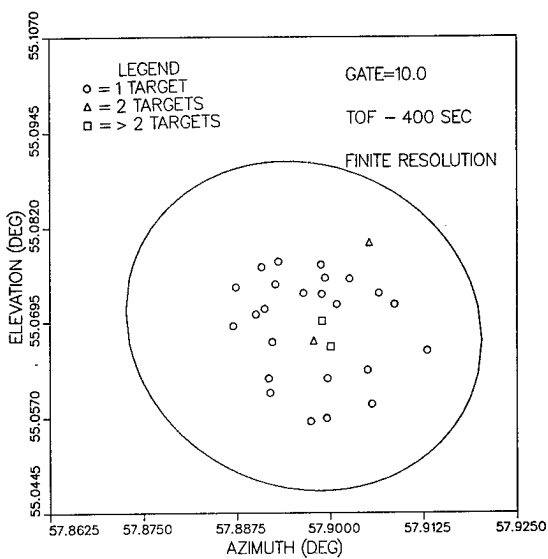


Figure 10. Gating at 400 sec for Finite Resolution Case

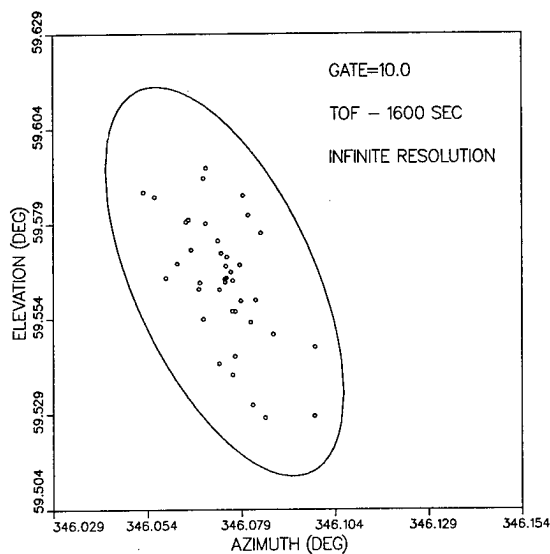


Figure 11. Gating at 1600 sec for Infinite Resolution Case

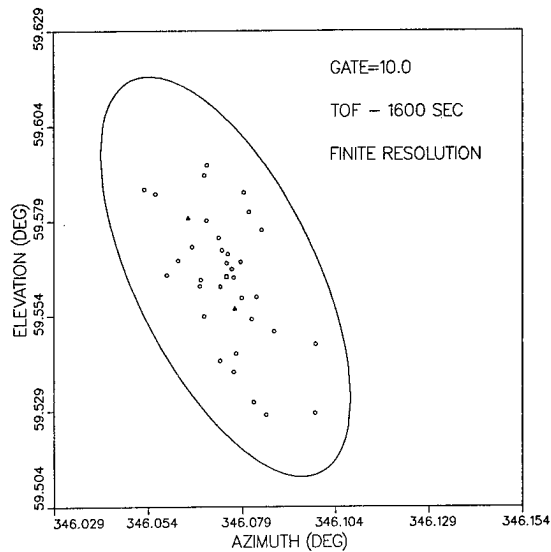


Figure 12. Gating at 1600 sec for Finite Resolution Case

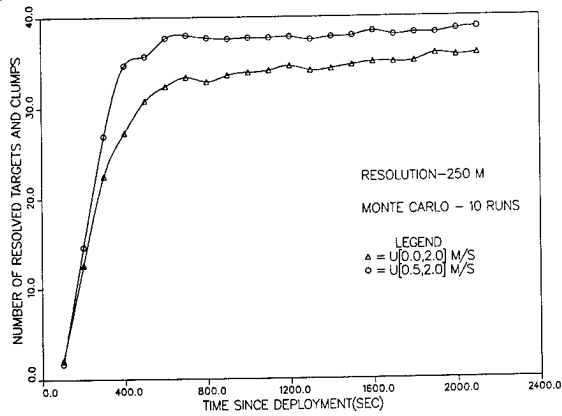


Figure 1. Resolution History for a 40 Target Cluster

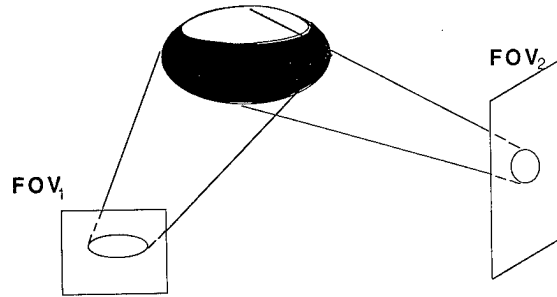


Figure 2. A Target Cluster as seen by Two Passive EO Sensors

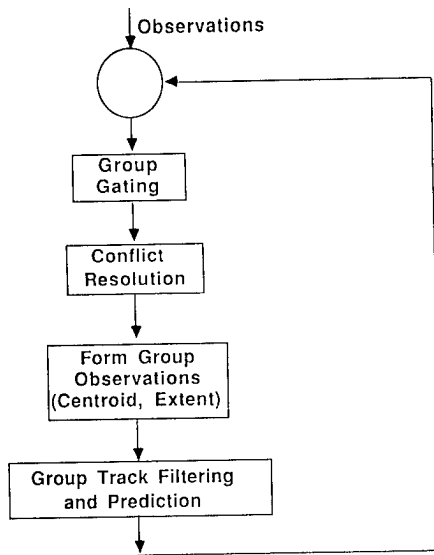


Figure 3. Group Tracking Flow Chart

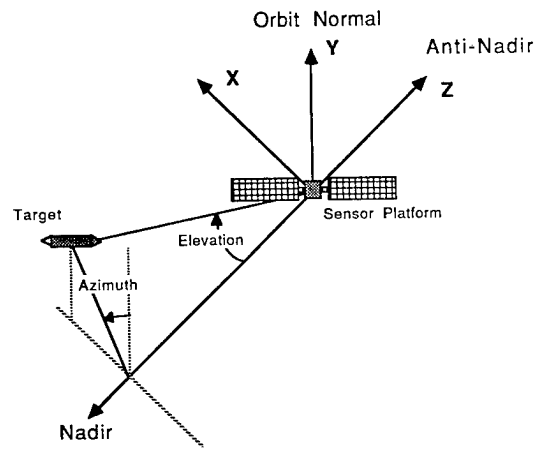


Figure 4. Sensor Measurement Geometry

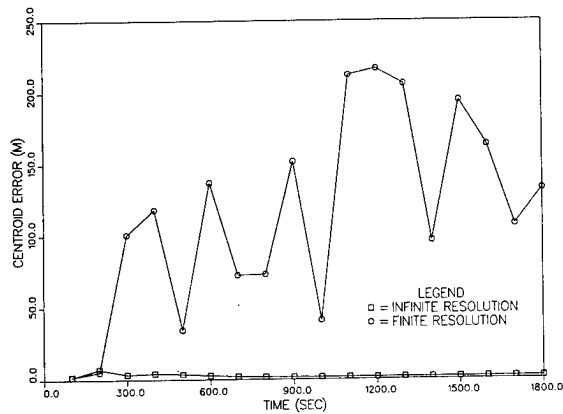


Figure 5. Centroid Position Estimation Error

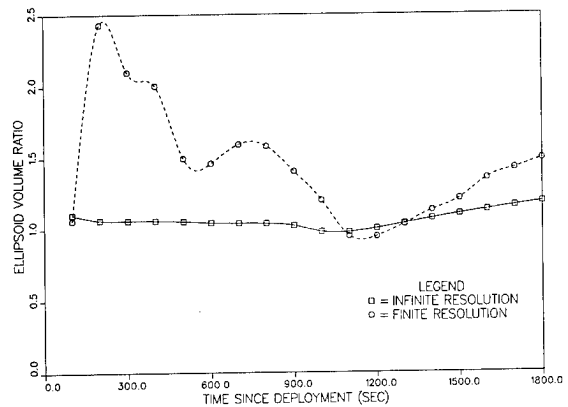


Figure 6. Ellipsoid Volume Ratio

COMMUNICATIONS IN COORDINATED TACTICS

Lee G. Morris

Veda Incorporated
600 Louis Drive
Suite 100
Warminster, PA 18974

Data fusion will establish synergism amongst sensors thereby extending the combatant's situational awareness. When the fusion concept is applied to multiple combatants in cooperative and coordinated tactics, it will suffer for want of communications.

Simply depicted, a combat platform consists of three basic systems; a weapons system, sensor system and command and control system. On a single platform, the unification of these elements enjoys advantages that are not available in a multiplatform cooperative effort.

What happens when the sensors, weapons and command and control are not collocated in a single platform?

The difficulty of correlating track reports within a Battle Force is well known. Two or three platforms perceiving the same target are likely to report two or three separate track positions due to several contributing factors.

Compensation for these errors can be established and eventually data fusion developed between platforms, but it requires extensive communications which will not be available in a hostile combat environment. The present state of communications development, as it relates to combat, is driven by a simplified threat which ignores the multifacets of communications in electronic warfare. This paper addresses what needs to be specified in tactical combat communications for supporting coordinated tactics in an electronic warfare environment.

COMMUNICATIONS IN COORDINATED TACTICS

Lee G. Morris

Veda Incorporated
600 Louis Drive
Suite 100
Warminster, PA 18974

INTRODUCTION

There has been considerable progress in the development of software systems supporting tactical "real time" decision making. Applications of these developments have primarily addressed the tactical decisions contained within a single platform. In the April 1988 issue of the **Naval Institute Proceedings**, CDR Kendall King described such a systems application for a ship board support system as shown in Figure 1. As a concluding sentence to this section of his article, he stated, "Capabilities could be expanded even more by communicating with tactical management information systems on other ships or at other activities via data links". The RDT&E thrust is now turning to expansion of cooperative or coordinated tactics, tying together many platforms and developing the synergistic effect of multiple sensors and weapons.

Simply depicted, a combat platform consists of three basic systems; a weapons system, sensor system and command and control system. On a single platform, the unification of these elements enjoys several advantages that are not available in a multiplatform cooperative effort. Several of the advantages are: (1) the bore sighting of sensors in relation to each other, (2) the direct alignment of the weapons to a host sensor integrator, (3) grid lock of targets and weapons to a single navigation reference, and (4) an internal communications capability to transfer massive amounts of information without concern of propagation constraints or intentional interference.

What happens when the sensors, weapons and command and control are not collocated in a single platform?

The difficulty of correlating track reports within a Battle Force is well known. Two or three platforms perceiving the same target are likely to report two or three separate track positions due to several contributing factors. For example: (1) the radar misalignment to the bore sight of an aircraft, compensated on the host but not

between platforms, (2) positional errors between the platforms navigation systems, (3) relayed track reports accumulate errors in position and time since track reports are not time tagged. These errors result in the rule of not shooting at a target not held on one's own sensor.

Compensation for these errors can be established and eventually data fusion developed between platforms, but it requires an extensive communications capability which is not presently available in a hostile combat environment. The present state of communications as it relates to combat, is addressing a simplified threat which ignores the multifacets of electronic warfare.

ELECTRONIC WARFARE

"Radio Electronic Combat (REC) is considered a primary weapons system by the Soviets. The Soviets have put immense effort and resources toward exploiting communications, radars, data links, and telemetry transmission for intelligence in peacetime. These same efforts will be used to direct targeting and for disruption in time of war." states LCDR Guy Thomas, USN, in **Military Electronics Countermeasures**, Dec 1982.

There are many books and articles addressing the Soviet REC which results in an awareness that jamming will take place in any conflict - including jamming of communications. The traditional response to this awareness has been one of specifying the need for an AJ data link to support the communications required for the single mission or a particular system. Communications so defined becomes an auxiliary to the capability being developed. This approach to an overall weapons system development cannot continue if communications is a vital need.

A very basic approach to antijam technology is to employ means to ensure a tolerable level of interference by forcing the tactical jammer to spread his energy over a wide portion of the radio frequency spectrum. To do this, the signal must be designed to take advantage of processing gain

(redundancy) or avoidance of the jammer (frequency hopping). To be effective, the redundancy or movement cannot be predictable to a hostile observer (encrypted).

This approach to AJ development seems simple enough, but it is only one of the parameters a communications system requires. A communications system requires an electronic counter-countermeasures (ECCM) capability. The electronic countermeasures (ECM) threat is not as simple as the "jammer". The ECM threat is a phased capability of exploitation.

1. First, an attempt is made to read the message traffic, if possible.
2. Employment of direction finding on signals and correlation of signal information is used for targeting.
3. Close examination of the signaling is used to establish the electronic order of battle for predicting operations.
4. If the system cannot be passively exploited, attempts will be made to spoof the communications system by false messages, capturing synchronization, or using other surreptitious methods to deceive.
5. Next, the destruction of the user is pursued by actually targeting weapons on the signal emissions.
6. Interference is further expanded by jamming on observed weakness like exposed synchronization signals, too few frequencies, etc., and,
7. Finally, barrage jamming is used as a last resort.

The communications that is to take place is command and control, voice and data. In cooperative engagements, there is coordination required between several combatants to exchange intelligence and status, structure the timing of events, and assign roles for action. This is not a one-to-one bulk data dump as can be found in telemetry links. What is required is an ECCM communications system meeting many constraints.

ELECTRO-MAGNETIC SPECTRUM

Returning to the earlier expression of what an AJ capability means, the use of the redundancy or agility in reducing the jamming effects, one must address the impact on the frequency spectrum. Figure 2 is a simplified depiction of the impact of AJ on the UHF spectrum specifically. Frame (a) shows the traditional use of this portion of the spectrum. Channels of the band (50 to 100 KHz) are assigned to operational users as nets or circuits and are dedicated continuously in time until the communications plan is changed. Since the channels are fixed in time and frequency and the bandwidth assigned is close to the data bandwidth required, the jamming signal can easily be assigned to the channel and its total energy matched to the signal energy as shown in frame (b). To defeat this jamming strategy, the redundancy is provided by frequency or time expansion and the agility is provided by rapid channel switching as shown in frame (c).

What becomes apparent in frame (c) is that the UHF spectrum, shown in this example, has been reduced from four voice circuits to two voice circuits. Of course the real world problem is much larger than this example. This shows a reduction of 2 to 1, where the actual need for redundancy requires a much, much greater reduction.

The way we use the frequency spectrum has been wasteful. The assignment of continuous time does not reflect the actual time the signal is on the air. Matching time of the signal (signaling rate) to the assignment time on channel has resulted in recapturing a number of the "circuits" that can be serviced. This is called Time Division Multiple Access (TDMA) technology, frame (d), and is commonly seen in communications satellite usage where the information throughout is limited. The problem is that the 225 to 400 MHz UHF military band is not so divided and the TDMA technique is not spectrum compatible with the way the channeling is assigned.

This UHF band is a small portion of the overall frequency spectrum and intuitively one would think there must be space available that is not structured adversely to employing ECCM techniques. Figure 3 shows the spectrum usage by Navy aircraft only. What is not shown, and what makes the problem more difficult, is the use of the spectrum by other services, other nations and civil facilities.

There are other spectrum parameters for ECCM communications that can be addressed by observing Figure 3. Consider the frequencies below 30 MHz. These frequencies are used primarily for extended or long range communications between net members, but they can only support mid-range data rates. Because of the long range attributes of these frequencies, they greatly abet the enemy's ECM exploitation by allowing direction finding, targeting and weapons direction. Next, look at the frequencies above 2 GHz. In this region, the range of propagation becomes very short and high power transmissions (high power generation) are required to overcome the noise background and high attenuation encountered. Usually, very directional antennas are employed to attain the required Effective Radiated Power (ERP) in the signal. These frequencies can support higher data rates, but the directionality required does not support tactical combat communications, command and control, which is omnidirectional in nature (broadcast vs. telemetry). More simply stated, pointing of antennas does not fit the operational connectivity requirements.

What is left are the bands shown in Figure 4 from 108 MHz to 1.5 GHz. As can be seen, fixed frequency assignments are high in this region. The Joint Tactical Information Distribution System (JTIDS) has taken advantage of the available time domain in the Lx-band by employing a pulsed waveform and TDMA. This has allowed the introduction for ECCM communications in the traditional TACAN/IFF band. The procedures for making maximum use of the time-bandwidth spectrum requires still further development.

ECCM AND COMMUNICATIONS

So far, this discussion has adhered to the simplified explanation of AJ and its impact on the frequency spectrum. In order to address all the needs in defeating the ECM exploitation, one must compare the conflicting needs for both ECCM and communications.

There are certain attributes of a communications system that the user takes for granted. When the communications requirements were not so great, relying on teletype and voice, there was no need to specify certain parameters of the platform system - they were expected.

What happens to these common place requirements when ECCM is introduced with its attributes of jam resistance, cryptographics security, and low probability of exploitation?

Multifunction operations from a platform were handled by multiple radios, each switched in as needed or dedicated to a particular function such as intercept control.

With the advent of ECCM, multifunction operations cannot be easily satisfied by multiple ECCM radios. To attain AJ, security and LPE, it is necessary to treat the radio frequency (r-f) spectrum as a system, the protocols of access and sharing are addressed at this spectral level, not the platform or user level. Under this approach, there is a platform terminal that must provide simultaneous service to the multifunctions (track reporting, sensor coordination, platform control, voice). Multiple terminals on a platform are not practical. First, true ECCM is expensive and not so readily afforded as the simple UHF radios. Solving the spatial r-f compatibility issues naturally results in a single terminal (various sizes) per platform.

Multinetting was handled by preset channel changes (20 presets per airborne UHF radio) switching from net-to-net as the action unfolded. Operating in two nets simultaneously was solved by having two radios.

Multiple netting in ECCM becomes complex. User profiles have changed. Aircraft, such as the F/A-18 and E-2C, and ships, such as AEGIS, have become multimission platforms requiring operations in autonomous networks supporting the various warfare areas. The AJ spectral concerns of ECCM communications are still prevalent in meeting these netting needs. The system must be organized on a Battle Force or theater basis with each platform having access through a single terminal to the spatial communications system.

Connectivity was one of broadcast. Anyone on the circuit or net could read the transmission of all others.

Connectivity is now complicated by the crypto keys used for randomness of signals and isolation of information on the need to know basis. By careful partitioning of circuits, the user need only observe (decipher) what is of interest to him. Not all information in the circuit or net need be filtered to read what is pertinent to each user. This partitioning also allows various levels of classified traffic to be supported by the same system and terminals.

Capacity was usually sufficient and any conflicts in use could be met with common user protocol of waiting until the channel was free.

Capacity must now be considered on a system basis, much like satellite communications where the repeating bandwidth is limited. As shown previously, the AJ aspects have reduced the throughput of the r-f spectrum. To increase the capacity, we must now employ time division multiple access techniques that again requires overall systems control on a spectral basis. For low probability of exploitation reasons, this access again must appear random lest the perceiver predict the EOB and then know when and how to jam.

Synchronization was not even a concern with the modulations employed. The initial problems requiring some sort of synchronization had to do with determining which was a "one" or "zero" in pulse code modulation. A simple synchronization preamble solved the problem.

Synchronization is now the key to accessing the ECCM system. Since the communications hinges upon a pseudo-random key stream, there must be a way to acquire this key stream. Further, this means of acquisition must be heavily protected from the LPE techniques which are looking for predictable aspects of any system. This is expensive to attain. It is the most vulnerable aspect of ECCM and cannot be treated lightly. Once again, this hinges upon the crypto codes and reduces the universality of access to the system.

Ranges required for tactical operations now definitely requires relay. Relay of course is replication of the circuit or channel which in ECCM requires another set or access parameters be added to the system. The circuit to be relayed must be received and then reconstituted for transmission on a separate circuit which impacts on system capacity.

With ECCM communications, the spectral r-f compatibility has been addressed earlier. With the design of a good ECCM system, platform r-f compatibility is part of the single terminal design and the biway operations with existing conventional communication equipments. With a poor ECCM design, the platform compatibility problem is virtually unsolvable on a fleet or theater basis.

Joint operations is still an operational command and control problem, even with identical ECCM or conventional equipments. ECCM does require greater planning since spatial operational netting and cryptographic codes must be predetermined and delivered. Spontaneity of operations is not possible, otherwise the system violates aspects of LPE.

SUMMARY

In summary, this paper presents only a cursory description of a complex issue that must be addressed in the design stage of systems. When planning systems that pursue cooperative tactics, the design of the information transfer system is no trivial task. As shown here, the information transfer problem is large and complex, while

the r-f spectrum available is small and crowded. JTIDS, with its attendant TADIL J, is the only system under development that addresses the command and control requirements of tactical combat. Each platform developer or manager cannot design his own communications, command and control system nor can the C³ system be designed remotely. The communications system now becomes an integral part of each platform design. The systems engineering aspects of the problem requires the cooperation of all users. This creates a major difficulty for communications system development. Addressing the communications problem brings along the realization that part of the programs development must be given up to another or collective authority. As a result, the issue of communications is often put off and not considered until the program is completed. Of course, at that time it is too late and the system, if fielded, relies on vulnerable communications.

Realization of the complexity of the issue can possibly lead to solutions: ignoring the issue will likely lead to failure.

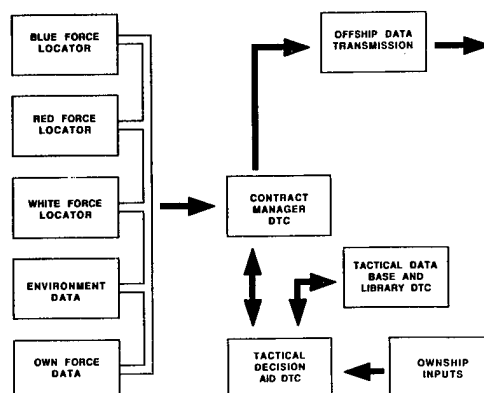


FIGURE 1
TACTICAL MANAGEMENT INFORMATION SYSTEM

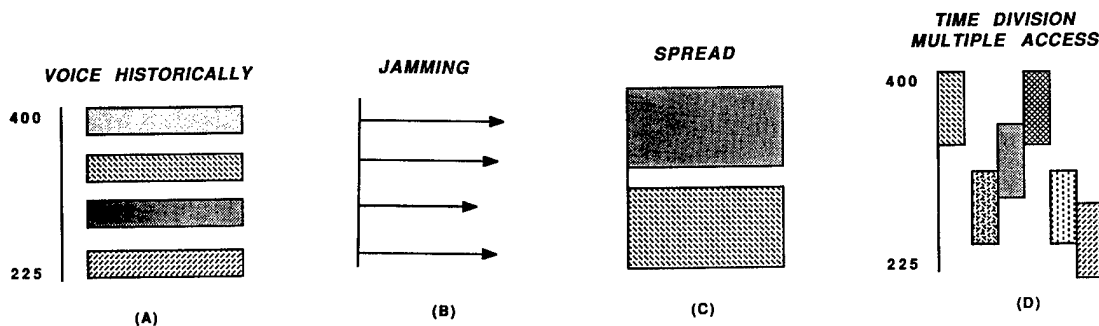


FIGURE 2
SPREAD SPECTRUM

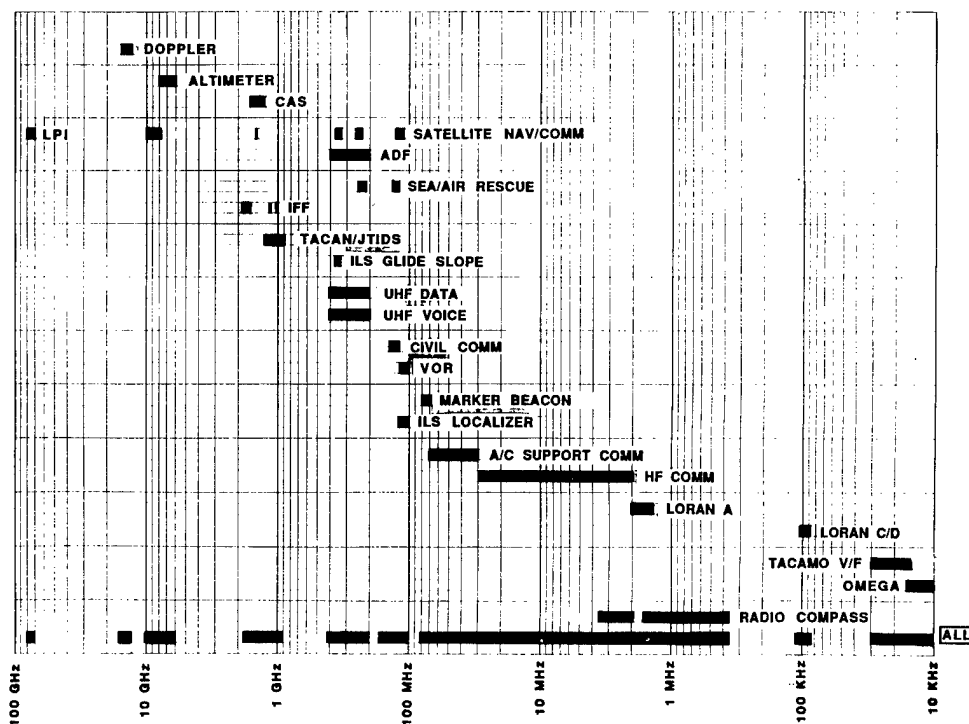


FIGURE 3
R-F SPECTRUM

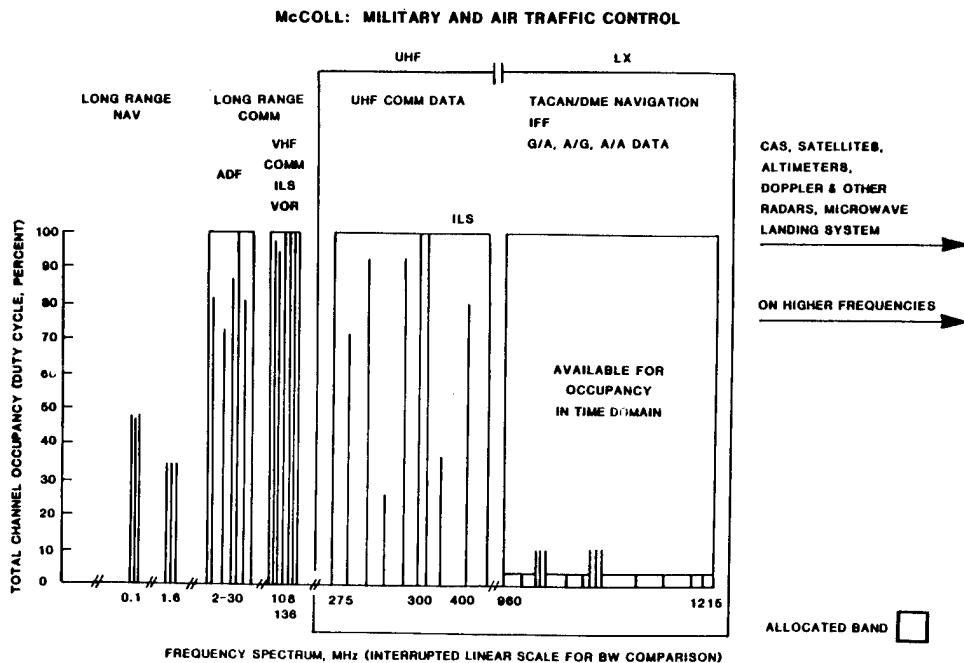


FIGURE 4
COMMUNICATIONS, NAVIGATION AND BEACON
FREQUENCY UTILIZATION

**ELECTRONIC WARFARE SITUATION ASSESSMENT PROCESSING REQUIREMENTS
FOR FIGHTER AIRCRAFT OF THE 1990's**

Ronald M. Yannone

**GE Aerospace
Aerospace Electronic Systems
Utica, NY 13503**

ABSTRACT

Electronic warfare (EW) systems aboard the fighter aircraft of the 1990's will require numerical and symbolic data processing to ensure platform survivability. This paper discusses an overview of the required algorithm which the EW Situation Assessment function will perform, its interfaces with the multisensor Fusion function and the Resource/Response Management (R²M) function, and threat data support needs. Specifically, the following algorithm requirements will be discussed: Threat Inference, Intent, Lethality, Countermeasures (CM) Effectiveness, and Risk. Threat Inference augments the output provided by Fusion in three areas: threat Identification/Classification (ID/Class), range, and the association of Fusion tracks as a weapon system. Threat intent uses kinematics, mode changes, and multispectral sensor report information to discern threat goals. Threat Lethality uses a priori and real time Fusion outputs to estimate the probability of kill with the associated time-to-go. Countermeasures Effectiveness factors in the countermeasure schedule from the R²M function and threat behavior as a result of applied countermeasures. Threat kinematics, emitter mode dynamics, and a priori countermeasures effectiveness are used to evaluate the reduction in threat risk. Threat Risk is shown to be a nonlinear function of Intent, Lethality, and Countermeasures Effectiveness. The paper describes how Kalman filtering and EW Expert rules-of-thumb comprise the basis of performing EW situation assessment.

ELECTRONIC WARFARE SITUATION ASSESSMENT PROCESSING REQUIREMENTS FOR FIGHTER AIRCRAFT OF THE 1990's

Ronald M. Yannone

GE Aerospace
Aerospace Electronic Systems
Utica, NY 13503

INTRODUCTION

Electronic warfare (EW) systems aboard the fighter aircraft of the 1990's will require numerical and symbolic data processing to ensure platform survivability. The charter of EW is to maximize mission success (P_{MS}) for the pilot while minimizing his probability of being shot down (i.e., maximizing probability of survival, P_S). This amounts to both a defensive as well as an offensive role. By defining threat and target, the roles of EW situation assessment become clear. A **threat** is the entity which is attempting or has the capability of doing harm to the ownship. Consequently, the ownship takes on a defensive posture to maximize P_S . A **target** is an entity which the ownship will attack or be offensive toward.

From these definitions, it is readily apparent that the use of entity (threat and target) location, Identification/Classification (ID/Class), and emitter mode dynamics is vital to perform EW situation assessment. The feedback to situation assessment from resource/response management must also be input to assist in closing the data processing loop when a response is made.

EW SITUATION ASSESSMENT REQUIREMENTS

Figure 1 depicts the generic data processing control loop aboard the aircraft. The inputs to situation assessment include target kinematic (position, velocity, and acceleration) and attribute (ID/Class, mode) information.

The emissions control (EMCON) posture aboard a fighter may be active, semi-active, or passive during different phases of a mission. As such, the output from fusion may not be as complete during all important phases of the scenario.

Consequently, an inference function to fill in the important missing data is required. Once the fusion track file has been augmented, the intent of the entity needs to be determined (whether it is hostile, non-hostile, or in a transitioning process). Then, given that the threat's intent is non-zero, a measure of its ability to harm the ownship needs to be evaluated. Then, given that the intent and lethality of the threat were assessed, a method to compute the threat risk using these data is required. Finally, as countermeasures are applied against the threat, a method of monitoring its effectiveness is required to factor this into the risk calculation reduction achieved. If the countermeasure is effective, the threat risk is reduced to an acceptable level. Otherwise the Resource/Response manager needs to schedule alternate assets and/or alert the mission computer. Figure 2 captures these required EW situation assessment functions in the form of a functional block diagram.

EW SITUATION ASSESSMENT SUBFUNCTION REQUIREMENTS

To develop situation assessment requirements, the EW expert rules-of-thumb need to be kept in mind. These involve threat data such as weapons effectiveness zones, multispectral emitters, and dynamics of the aircraft and missiles during an engagement.

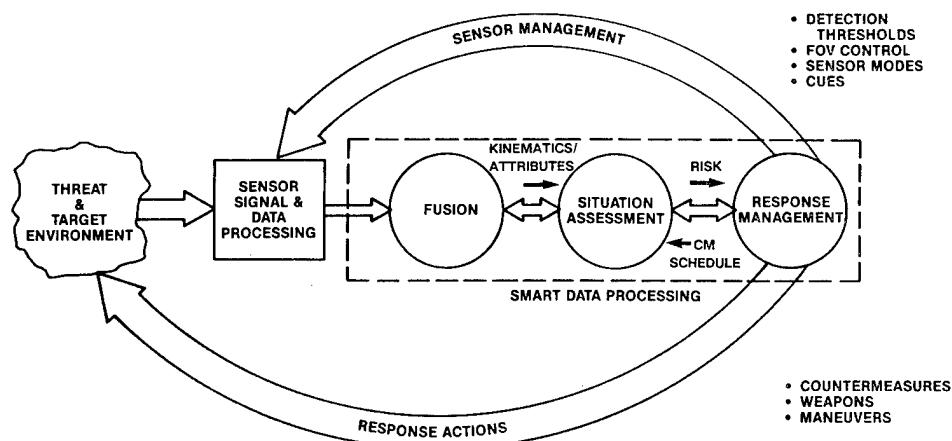


Figure 1. Responses are Based on Continuously Updated Awareness of Total Environment Situation

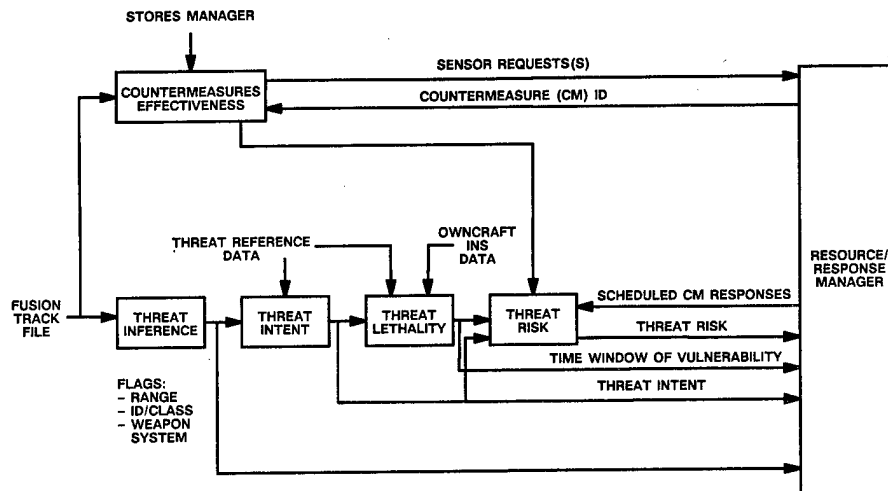


Figure 2. EW Situation Assessment Block Diagram

Threat inference needs to augment the fusion track file in range, ID/Class, and weapon system categories.

Range inference is performed to supply range information to a threat, be it airborne or land-/sea-based. Methods to do this vary as functions of geometry and assumptions made.

ID/Class inference supplies some form of ID/Class of the entity when no real time sensor information is available. Use of threat data relative to the mission provides clues as to probable threats that are likely to be encountered.

Weapon system inference will thread different fusion track file entries which do not "appear" to go together based on normal fusion rules. This should cover multispectral emitters belonging to a weapon system. A weapon system will include, in the case of an aircraft, the various emitters onboard.

Threat intent examines real time fusion track file data which provides clues as to the intent of the threat to harm the ownship. It will piece together the kinematic history (aspect of the threat relative to the ownship in the case of an airborne threat), the RF emissions (ID/Class and node), and the complementary multi-spectral weapon system components.

Threat lethality evaluates the ability of the threat to kill the ownship given that its intent is nonzero. It will need to use threat platform ID/Class and state vector information to compute probable weapon(s) to be launched, their flyout characteristics, and the assessed resultant PKILL given the fact that a weapon was used.

For land/sea threats, the cumulative PKILL must be determined based on ownship speed, altitude, and heading relative to the netted threat laydown forseen to be encountered.

Countermeasures effectiveness will factor in the threat and applied countermeasure(s) to assess whether threat risk has indeed been reduced by using fusion output and countermeasure ID.

This is assessed by monitoring RF emitter modes and kinematics of missiles in flight. For the missile case, some form of Kalman filter design is required as outlined in Figure 3.

The key role of the Kalman filter in EW situation assessment resides in countermeasures effectiveness. The modeling of missile dynamics affects state vector estimation accuracy. The Singer [1] model was explored and is summarized in Figure 4.

In this model, the target acceleration $a(t)$ is modeled as a zero-mean random process with exponential autocorrelation. In Figure 4, σ_m^2 is the variance of the autocorrelation and $1/\alpha$ is the time constant of the target acceleration. The target can accelerate at a maximum rate A_{MAX} ($-A_{MAX}$) and will do each with a probability P_{MAX} , and will accelerate between the limits $-A_{MAX}$ and A_{MAX} according to the appropriate uniform distribution. For missiles, this model was modified to accommodate the non-linear velocity versus time profile as shown in Figure 5.

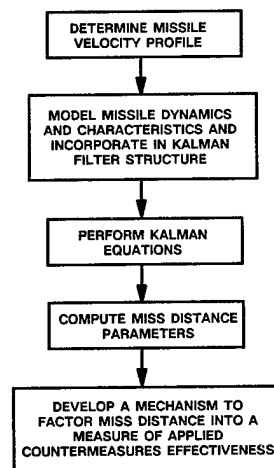


Figure 3. Missile Endgame CM Effectiveness Incorporates All Available Sensor Kinematic and Attribute Information

Using the modified Singer model has led to Kalman filter designs which do not diverge.

Threat risk is a combination of the threat intent, lethality, and countermeasures effectiveness. The manner in which to combine these parameters may vary. One way to approach the calculation follows.

When the threat intent, lethality, or both is zero, then the risk should also be zero. If the intent and lethality are non-zero, then a product of the two is a way of combining the two. So, threat risk without countermeasures applied can be written as a product of intent and lethality. When countermeasures are applied, the EW system has effectively closed the loop against the threat. Consequently, the risk without applied countermeasures needs to be attenuated accordingly. This leads to the following expression for threat risk:

$$\text{Risk} = \text{Intent} \times \text{Lethality} \times \text{CM Effectiveness} \quad (1)$$

SIMULATION RESULTS - AIRCRAFT AND MISSILE SALVO CASE

Three entity types are tracked: an advanced air-to-air missile, surface-to-air missile, and threat aircraft (which fires the air-to-air missile).

The engagement is a low altitude scenario slice where the AAM and SAM form a salvo against the ownship.

Figures 6, 7, 8, and 9 give normalized velocity and range data for the air-to-air missile (AAM) and surface-to-air (SAM) missile threats, respectively.

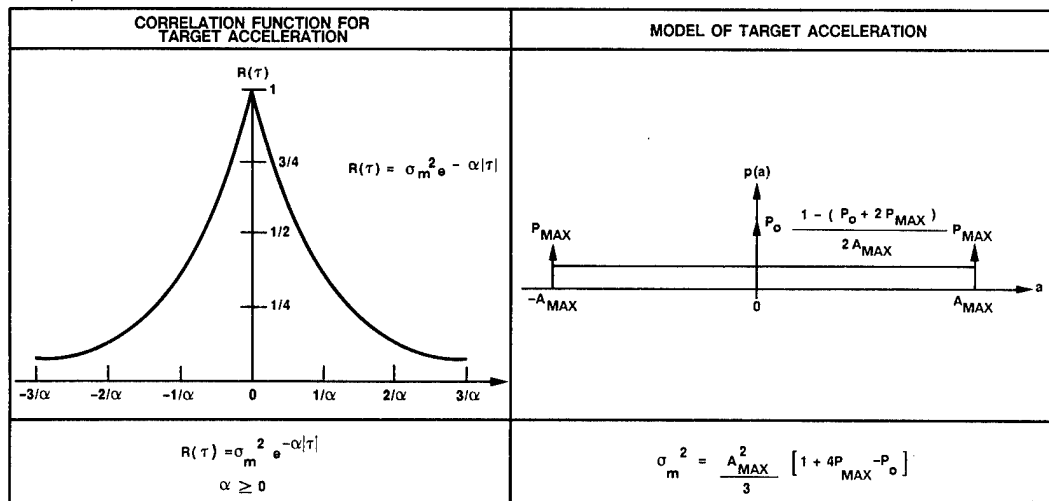
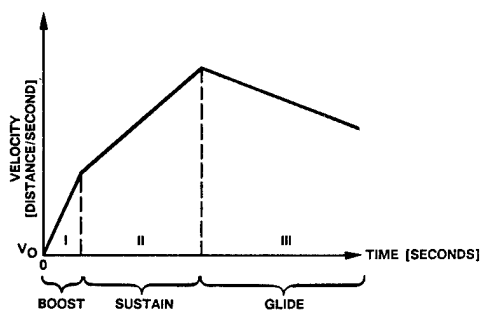


Figure 4. Singer Target Acceleration Model



When a missile launch is confirmed and possible missile ID (SA-XX or AA-YY) has been established, an accurate Kalman filter model can be performed by using the a priori phases (I, II, III) of missile flight. The accelerations and time boundaries then prescribe accurately the first-order Markov parameters.

Phase I - $P_{max} = 0.98$, $P_0 = 0.00$, $A_{max} = A$ g's, $\alpha =$ proportional to boost time
Phase II - $P_{max} = 0.98$, $P_0 = 0.00$, $A_{max} = B$ g's, $\alpha =$ proportional to sustain time
Phase III - $P_{max} = 0.98$, $P_0 = 0.00$, $A_{max} = C$ g's, $\alpha =$ proportional to glide time

Figure 5. Three-Stage, First-Order Markov Missile Acceleration Model

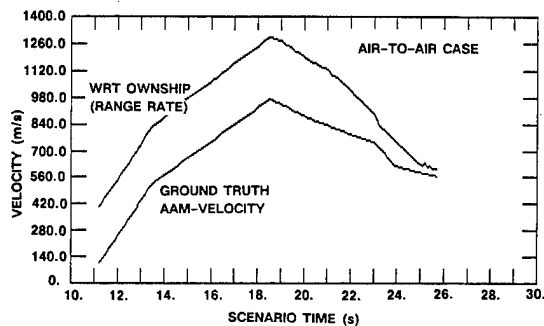


Figure 6. Ground Truth of AAM Velocity and Range Rate versus Time

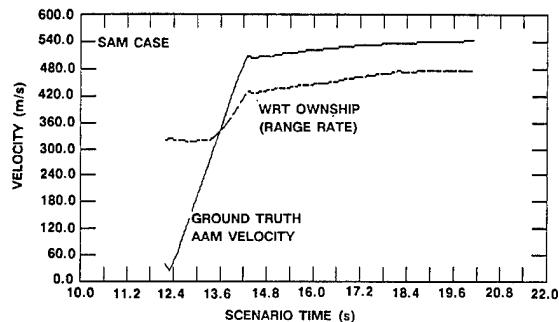


Figure 8. Ground Truth SAM Velocity and Range Rate versus Time

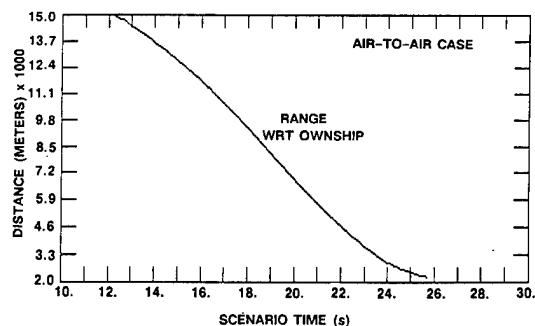


Figure 7. Range versus Time for AAM/Ownship

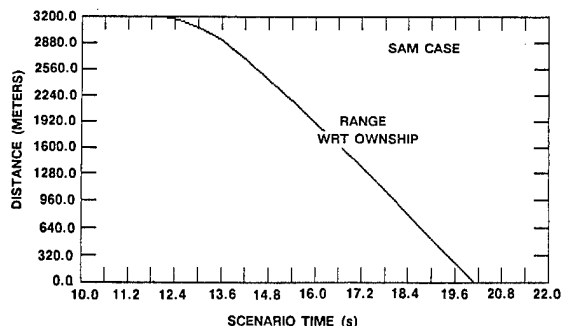


Figure 9. Range versus Time for SAM/Ownship

Algorithms not discussed in this paper were used to implement Inference, Intent, Lethality, Countermeasures Effectiveness, and Risk, and were applied to this scenario. On-board sensors were selected and countermeasures were applied for the scenario.

Figures 10 and 11 depict the risk parameters for the AAM and SAM.

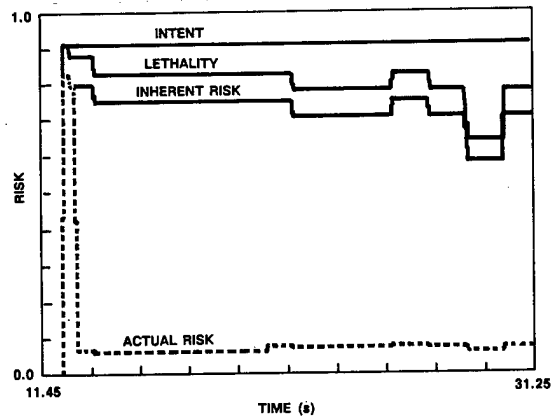


Figure 10. Components of Risk: AAM

SUMMARY

In this paper we have done the following:

- Discussed the role of EW situation assessment (SA) in the fighter of the 1990's.
- Walked through the top-level SA requirements which need to be performed by situation assessment.

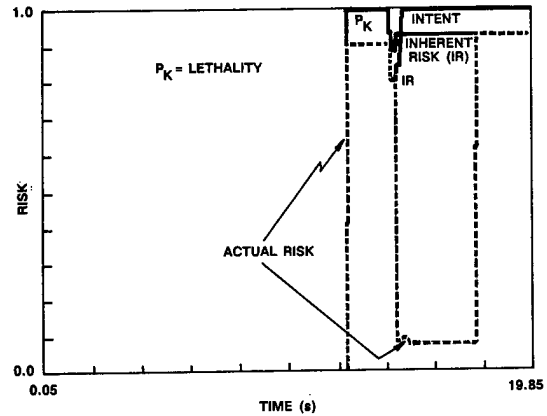


Figure 11. Components of Risk: SAM

- Mapped the top-level SA requirements into subfunctions, their requirements and interfaces between fusion and resource/response management.
- Discussed how Kalman filtering and EW expert rules-of-thumb comprise the basis of performing EW SA.
- Showed typical risk parameters via use of simulation results for AAM and SAM threats.

REFERENCES

- [1] Singer, R.A., Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets, *IEEE Transactions on Aerospace and Electronic Systems*, July 1970, pp. 473-483.

THIS PAGE LEFT BLANK INTENTIONALLY

IDENTIFYING ALIEN CONTACTS
IN BAYESIAN CLASSIFIERS

D. G. Shankland, J. Johnson, D. W. Princehouse, & M. R. Smith

Boeing Computer Services
P. O. Box 24346
Seattle, WA 98124-0346

A statistic has been devised which indicates, with reasonable probability, when a contact which has been identified as some particular member of a possible contact list by a Bayesian classifier is in fact not a member of that list at all, i.e., is "alien" to the database at hand.

Relying on one's a-priori knowledge of the statistics of the observation process, the statistic is the average, z , of N mean-zero, unit variance statistics, where N is the number of observations from the various sensors employed. If the contact has parameters differing sufficiently from the member of the set which was chosen by the Bayes' classifier, then the z -value will diverge from zero beyond the range commensurate with the number of observations, and the operator is warned of the presence of a new contact not in the existing list.

The statistic is linearly related to the conditional entropy of the stream of observations, and combines the outputs of an arbitrary number of sensor inputs. In simulation studies with four "sensors", it correctly indicated contacts as alien 58% of the time, while making only 2% errors in identifying contacts which were in the database.

IDENTIFYING ALIEN CONTACTS IN BAYESIAN CLASSIFIERS

D. G. Shankland, J. Johnson, D. W. Princehouse, & M. R. Smith

Boeing Computer Services
P. O. Box 24346
Seattle, WA 98124-0346

INTRODUCTION

The Bayesian classifier is generally regarded as the algorithm most efficient at identifying a contact as a particular member of a list of possible contacts, given observation data with some known statistical distribution. However, there are (at least) three conditions which can cause difficulties with the process, namely

- a) poor knowledge of the statistics
- b) missing data on some of the characteristics of some of the list members
- c) a contact that is not in the list at all ("alien")

This paper will address the third difficulty. It will outline the computation of a statistic which can indicate the presence of an alien contact by quantitatively comparing the stream of observations with one's expectations for the list member which the Bayesian classifier selects.

This will enable Bayesian classifiers to be used with more confidence, exploiting their simple treatment of sensor information, and will develop in trained observers an additional feel for the likelihood that the contact is unlike previously encountered ones. The implementation of the method is straightforward, although it does rely more heavily upon accurate knowledge of the statistics of sensor performance, pointing out again the necessity for making measurements and historical records of essential sensor statistical parameters.

The first section will treat the behaviour of the product of the conditional probabilities of sensor outputs, developing the importance of the logarithm of the conditional probability as a random variable of key importance in the alien identification process. The second section will discuss methods for computing means and variances of these logarithms for various sensors, in order to center and normalize the measured random variable into the "z-statistic", a

mean-zero, unit variance random variable which can be combined from multiple sensors into an averaged "z" which can then be quantitatively examined to obtain the warning of alien presence. The third section will present some test results, and discuss the performance of the method. A summary will follow.

A STATISTICAL ESTIMATOR FOR AN ALIEN SHIP

The initial assumptions are that while the ships in the database, and the contact, have unique, well-defined properties, the observation is necessarily corrupted by transmission losses, observational inaccuracies, and the like, so that what is reported is likely not to be the exact truth, but only something nearby. One also assumes that one has enough information about the performance of the observational system so that one can calculate the probability that some observation "o" might be reported, given that the contact was actually ship "s", $P(o|s)$.

In particular, the paradigm is, say, a measurement of the location of a ship's mast, as a percentage of the ship's length. Truth for a given ship is x_s , and the observation is reported as y . One assumes that one can calculate the probability $P(y|x,dx)$ which is really, given the discretization of one's reporting at intervals of, say, $dx = 1\%$, the probability that the report will be in the range $[y - .005, y + .005]$. For the discussions here, simply assume that the probability density is Gaussian, with a mean of x and a standard deviation of s , so that the probability " $P(y|x,dx)$ " is

$$P(y|x,dx) =$$

$$dx \cdot \exp[-0.5 \cdot (y-x)^2 / s^2] / (s \cdot \sqrt{2\pi}) \quad (1)$$

The actual probabilities are computed according to the database information for the particular observation system and variable considered, but do not change the considerations involved here.

The Bayesian approach regards the observations as necessarily coming from some ship in the database, the only task being to assess the relative likelihoods of their having come from each ship. This information is then combined with the prior probabilities of the various ships to yield the final probability that the contact is any particular ship in the dictionary, given the observations:

$$P(\text{ship}|\text{obs}) =$$

$$P(\text{obs}|\text{ship}) * P_0(\text{ship}) / P_0(\text{obs}) \quad (2)$$

The $P_0(\text{obs})$ is determined simply by normalization, since one knows the probabilities in (2) must sum to 1.0.

However, in the case where there may be ships not in the database, this determination of $P_0(\text{obs})$ is necessarily incorrect. Yet the Bayes' procedure will select a ship (or ships) from the database as the most likely, and give one no warning that there is really something out there that has not been seen before.

So, consider the case where the Bayes' process has narrowed down the choices to only one ship in the database as the likely contact, because, of the ones available to us, it was consistently most like the observation series. Then one could ask, "If ships may exist which are not in the dictionary, is the observation series 'o' probable, given that the contact was the ship 's'?" I.e., on an absolute basis, how big is $P(o|s)$?" That is, if ship "s" were the contact, is the probability $P(o|s)$ one in a thousand? in a million? Surely there is some value of $P(o|s)$ at which one would be willing to assert that, even though "s" looked most like "o" amongst those in the dictionary, it didn't look enough like it to be satisfactory and one is led to the conclusion that there must be something new out there.

Unfortunately, even if the contact were "s", as each of a sequence of observations is received, the associated probability bounces around. If one multiplies those probabilities together (assuming the observations were independent) the overall probability of the sequence gets smaller and smaller, and one is led to trying to evaluate whether or not it is getting smaller at the right rate. How can this be done?

If one had a single fraction q such that, on the average, q were equal to the product of N observed probabilities, then for the i -th observation o_i , one could take the ratio r_i of $P(o_i|s)/q$, and in some sense it should be about one. The correct sense is that

$$\lim_{N \rightarrow \infty} [r_1 * r_2 * \dots * r_N]^{(1/N)} = 1 \quad (3)$$

If, however, the observations were coming from a ship not in the database, which had a different value of "x", the probabilities would be consistently low, and the product of the ratios would tend rapidly to zero. So the strategy is clear: one should find the number "q", and upon receiving each observation o_i , compute $r_i = P(o_i|s)/q$, multiply the accumulated product (initialized at 1.0) by r_i , and if the result is smaller than some tolerance,

declare that the putative ship could not be the actual target so that there must be something new out there.

Note that, if the distributions of the observations are different for the ship "s" and the unknown contact, it is possible for the ratio to be greater than 1.0 on the average, and the product to diverge to large numbers. This would happen, e.g., if the two ships, "s" in the database and "c", the contact, had the same value of x but c 's distribution were much narrower. Then on computing $P(o|s)$, one would be getting an excess of large probabilities, and $E\{r_i\}$ would be greater than one. This probably will not happen in our case, since the distribution is presumably characteristic of the observation system and not the ship, and so would be identical for both "s" and "c". Then, if the ship parameters are the same, the two contacts will look identical to the system, and so no discrimination is possible. In a multi-sensor system, however, there will generally be different values of some variables, so that the ratio will still tend to zero for a non-database (alien) ship.

Operationally, one has difficulties with the product formulation of this sequential test, so instead one considers the logarithm of the product, since this is simply the sum of the logarithms of the individual ratios. The N -th root of the product is then the average, and "q" is given by the expectation of the logarithm of the probability,

$$\ln[q] = E[\ln[P(o|s)]] \quad (4)$$

which, apart from a constant multiplicative factor, is the conditional entropy of the observation stream.

The logarithm of the ratio r_i can then be normalized to a statistic z_i , where

$$z_i = k * [\ln[P(o_i|s)] - \ln[q]] \quad (5)$$

This is a mean-zero variable, and if k is chosen so that the variance is one, it can be averaged with other z 's (perhaps from different sensors) to create a sequential test statistic (the cumulative average) which tends to be Gaussianly distributed with mean zero and variance $1/N$. Thus one can place probabilistic bounds on the averaged z , and reject the hypothesis that the observations could have come from ship "s", with any desired degree of confidence.

It is instructive to compute q for the case of the multivariate Gaussian distribution. Let the mean and covariance matrix for the random vector \underline{x} be \underline{m} and B , and let the inverse of B be L . Then the statistic z becomes

$$z = 1.0 - k * (\underline{s} - \underline{m})^* L * (\underline{s} - \underline{m}) \quad (6)$$

where $(\underline{s} - \underline{m})^* L * (\underline{s} - \underline{m})$ is the Mahalanobis distance associated with the measurement \underline{s} . Thus, the central quantity z for Gaussianly distributed observations is nothing more than the mean of the Mahalanobis distances, and this is compared with the expected value of 1.0. As the number of observations N increases, the distribution of the mean z tends rapidly to a Gaussian, with a variance of $1/N$.

One's task, then, is to compute the mean $\ln[q_i]$ and standard deviations d_i of the various $\ln[P(o_i|s)]$'s, so that upon receiving a particular observation o_i , one can compute

$$z_i = \{\ln[P(o_i|s)] - \ln[q_i]\}/d_i \quad (7)$$

and average it with the previous ones, and test to see if that average is beyond the bounds expected for the number of observations gathered so far. If it is, one can safely drop that ship "s" from active consideration, and if there is no alternative, declare that the contact "c" is not in the database.

COMPUTATIONAL CONSIDERATIONS

It is clear that, for the identification of a contact "c" as not being a ship "s" that is in the database, one will need to compute the average of many samples of the quantities

$$z = k\{\ln[P(o|s)] - E\{\ln[P(o|s)]\}\} \quad (8)$$

where $P(o|s)$ is the probability of receiving the observation "o", given that the sender was ship "s", $E\{\cdot\}$ is an average over all the observations that might be received, and k is chosen so that the variance of z is one. Sometimes k and $\ln[q] = E\{\ln[P(o|s)]\}$ can be computed analytically, but more often, the distributions will be awkward enough that the computations will have to be done numerically. Fortunately, one needs only the first two moments of the variable $\ln[P(o|s)]$, so the numerical computation should not be too expensive. There are a couple of issues to consider.

First, let's examine the ideal situation, where one knows the distribution and can compute all the relevant quantities. Assume a one-dimensional quantity being measured, which for the ship one thinks it is is distributed normally about zero with variance one. Then, as has been previously derived, the z -parameter is related to the Mahalanobis distance, i.e.,

$$z = (1 - x^2)/\sqrt{2} \quad (9)$$

After averaging a number N of independent observations of z , one has a quantity \bar{z} with mean zero and standard deviation $1/\sqrt{N}$. So one is happy if, after Bayes selects a ship, the average \bar{z} for that ship is less in magnitude than about $1/\sqrt{N}$.

However, assume that the contact was really a ship for which the quantity being measured had a value of y , not zero, but (since the statistics are due to the measurement system) distributed with the same standard deviation. Then one finds that an average of N z 's computed from formula (1) will have

$$E\{\bar{z}\} = -y^2/\sqrt{2} \quad (10)$$

and

$$E\{\bar{z}^2\} = y^4/2 + (2y^2+1)/N \quad (11)$$

so that the variance of \bar{z} is $(2y^2+1)/N$. One wants to take enough observations so that the standard deviation of \bar{z} is smaller than its average value, or conversely, one can only discriminate against ships for which the y value exceeds some critical size y_{cr} for the number of observations that one has made. These imply that either

$$N > (4y^2+2)/y^4 \quad (12)$$

or

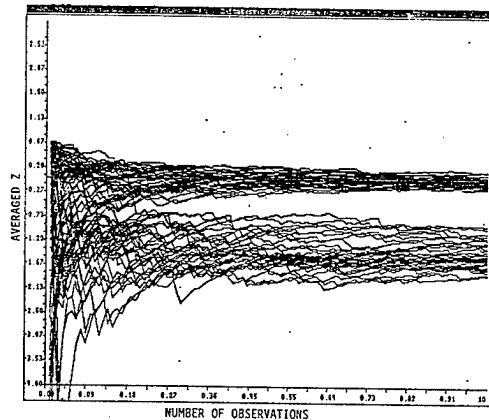
$$y_{cr}^2 > 2(1+\sqrt{1+N/2})/N \quad (13)$$

A little table of y_{cr} vs. N is enlightening:

N	y_{cr}
4	1.169
16	0.707
64	0.459
256	0.311

Thus, it is very difficult to resolve two ships whose values are separated by less than about a half of a standard deviation (William Strutt, Lord Rayleigh, was right again!). Experience with a simulation, so far, is that some ships are really not separated by that much, so that considerable mis-identification is possible without the z -parameter issuing a credible warning.

A simple simulation of the z -parameter computation for the case of a pair of contacts is shown in figure 1. It is obvious that, early on, there is insufficient discrimination between the two sets, but that eventually they coalesce into clearly distinct regions, one centered about 0.0 and the other about 1.4 (standard deviations). It would be highly improbable that one could mistake a trace from the second set as coming from the first, once sufficient observations had been made.



z -traces

figure 1

Unfortunately, one doesn't usually know accurately the probability $P(o|s)$. Also, the results are even more sensitive to the details of the distribution than in the operation of the Bayes classifier itself. There are two consequences of this:

The first problem is that many of one's model distributions are very complex. One must conclude that, except for finite-dimensional cases with known distributions and a few especially simple distributions such as uniform and Gaussian, analytic computation is useless, and so one needs a program that computes the mean and mean-square by a Monte-Carlo method.

If one has unclear knowledge of the distribution, then the actual z 's may well have a different distribution, with means

and variances both different. The variance error is not ordinarily serious since it just means one doesn't have accurate limits on the bounds for warning, and one could relax those bounds a bit to take care of the error. The error in the mean, however, is serious, as one might well have false alien warnings arising from one's computations. So one surely must provide for experimental data to replace the precomputed quantities. That means that, in the aircraft (or simulation thereof), real-time updates must be performed on the moments of $\ln[P(o|s)]$ so that a realistic database can be accumulated. The best way of doing this is not yet clear, but it seems as though some sort of autoregressive filter will allow a gradual replacement of theory by experiment. Then this will imply some database management problems to ensure data integrity and consistency.

The determination of the limits, within which the averaged z can lie without one's being concerned about the possibility that the contact is an alien, must be made only after considerable practical experience with actual data. Even then, the report must be probabilistic, and relates only to the probability that an observation sequence could have come from the Bayes-selected ship, not the probability that the contact is an alien.

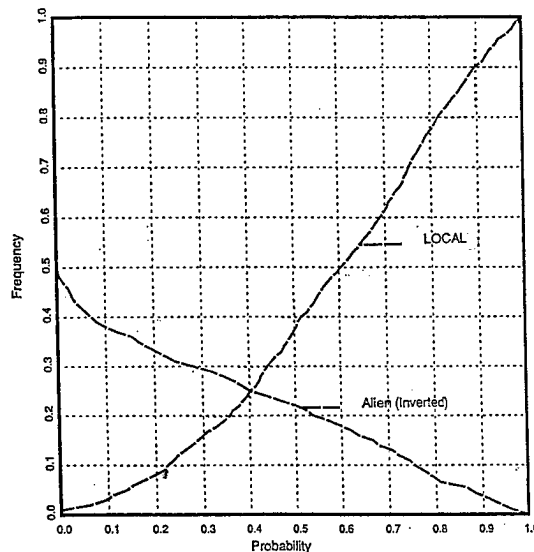
What one comes up with, in essence, is a self-adapting system which automates a selection, evaluation, and data-updating process, and incorporates operating experience in presenting probabilities for further operator evaluation.

A program, MONCAR, was created to enable large numbers of simulated ships to be identified and the z statistics generated, while using the expected values for $\log(p)$ and $\log(p)^2$ generated from a previous offline calculation. The purpose of this was to obtain insight of the value of using the z statistics in identifying alien ships. A switch was added that enables the operator to specify that the contact is "alien" or "local" (i.e., in the data base). If a ship is specified as an alien then it will be used to generate the observations, but will be omitted from the ships available for selection. For each value of the averaged z statistic a corresponding probability of its occurrence is generated. It is expected that most of the ships chosen when an alien is specified will have small z -probabilities, but if a ship is specified to be a local ship, then the chosen ship should usually have a large probability.

TEST RESULTS AND PLOTS

The expectation was tested by running a combination of four sensors with features consisting of radar antenna positions, mast height, mast position, and two superstructure profile heights as measured by a simulated ISAR, and ESM information. Each of a list of 100 ships was selected as local 10 times for a total number of 1000 trials. The z -probability data was sorted and plotted. Then the test was repeated with each ship selected as an alien. From the plotted data a probability threshold could be set for declaring a ship as a probable alien with a given confidence. For example, in figure 2, one can see that if one chooses to regard all ships with z -probabilities greater than

0.05 as local, and less than 0.05 as alien, about 2% of the time this will be an erroneous decision for ships that are in the database. However, about 58% of the non-database ships will be correctly identified.



z -probability frequencies

figure 2

SUMMARY

A statistic, z , the mean of the centered and normalized logarithms of the conditional probabilities of the observations in a sequence, has been identified as useful in discriminating between contacts in one's database and alien contacts. The computation of this statistic requires knowledge of the distributions of each type of observation, from which one can compute the logarithm of the probability of the sample observations, given the ship, as well as the mean and variance thereof. Upon identification of a contact by the Bayesian classifier, the probability of one's having accumulated that particular z is checked to see if it is within some preset tolerance, consistent with one's desired level of risk, and the contact is then reported as local or alien. In one series of simulations, some 58% of the alien ships were correctly identified as such, with only 2% of the local ships being misidentified. The z -statistic thus appears to be a useful adjunct to a Bayesian classifier, particularly when one wishes to limit the data base, yet be warned when a non-database contact is encountered.

THIS PAGE LEFT BLANK INTENTIONALLY

An Automated Tactical Situation Assessment Methodology: *Its Limitations and Impact Upon Tactical Command and Control*

Douglas Walter J. Chubb

U.S. Army CECOM Center for Signals Warfare
Vint Hill Farms Station
Warrenton, Virginia 22186-5100

Since 1984, basic research efforts have been ongoing to provide the U. S. Army Division/CORPS Staffs with automated assistance during the preparation of a tactical situation assessment (TSA). The current TSA research paradigm assumes that the generation of a TSA is a problem solving process wherein ongoing tactical plans are formulated and recognized. The TSA *tactical plan recognition* paradigm makes use of Wilensky's relational language, KODIAK, to express high-level, generic, enemy plans. Low-level KODIAK Actions are expressed using Schank-type scripts which are believed to be stereotypic and extensible, thus satisfying the research requirement for a *domain independent* and *extensible* TSA paradigm. This paper argues that the efficacy of the plan recognition methodology is, however, uncertain since each KODIAK Action script represents a highly abstracted tactical plan. The mathematical basis for a *plan structure* and a *plan abstraction transformation* is formally developed. It is shown that each KODIAK Action must be represented as either a deterministic plan or as the union of a finite number of deterministic plans. The paper concludes by examining the behavior of the TSA paradigm, assuming that the KODIAK Actions are improperly represented. It is shown, assuming a uniform probability of detecting *pre* and *post* Action states, that the TSA paradigm will tend to converge to either an unexceptional or worst case scenario. The impact of the described TSA methodology on the Army Command and Control process is examined, and suggestions for improving the plan recognition paradigm are offered. Finally, experimental evidence exists which appears to support the author's theoretical results.

An Automated Tactical Situation Assessment Methodology: *Its Limitations and Impact Upon Tactical Command and Control*

Douglas Walter J. Chubb

U.S. Army CECOM Center for Signals Warfare
Vint Hill Farms Station
Warrenton, Virginia 22186-5100

I. INTRODUCTION

Since 1984, basic research efforts have been ongoing to provide the U. S. Army Division/CORPS Staffs with automated assistance during the preparation of a tactical situation assessment (TSA). Both contractual and in-house research efforts have been involved in this effort. A "how to perform" TSA paradigm has been developed under U.S. Government contract by The Analytic Sciences Corporation (TASC). Work is presently underway to implement and test these ideas [Hatfield]. As with most new ideas, this paradigm has been received with some skepticism and criticism. Hopefully, this paper represents a part of that healthy process of peer review and, as such, is a critique of the project's *plan recognition paradigm*. Of particular concern to the author are the possible TSA analysis errors which may develop when improperly abstracted plans are used as a part of the automated TSA *plan recognition process*.

This paper begins with a brief discussion of how a rapidly changing technology has affected the TSA process during the 20th Century. This discussion concludes with a commentary on the problems currently facing a military Staff during the preparation of a TSA. Next, the mathematical basis for *plan abstraction* is developed. The TASC TSA *plan recognition* paradigm is described and critiqued. The paper concludes with a discussion of a possible situation analysis paradox which may result if improperly abstracted plans are implemented within the proposed plan recognition paradigm. The effects of this analysis paradox on the Army's Tactical Command and Control process are also examined. Finally, a number of possible improvements to this paradigm are described.

A. The Effect of Technology upon the Tactical Situation Assessment Process

Technology has been the driving force changing the methodology of warfare. Although the high-level objectives of war (e.g., to occupy and control) have remained essentially unchanged since Biblical times, the methods used to implement these tactical objectives have changed considerably. Technological innovation and growth continues at an exponential rate. The effect of technology upon the art of warfare has forever changed the TSA process in three fundamental ways.

First, and most importantly, technology has caused the TSA *processing time to shrink*. The maximum time permitted to react to a tactical threat has generally decreased with each military technological improvement. For example, during World War I global armies faced one another, deeply entrenched, along a slowly moving, imaginary line called a Front. In this seige-like condition, there was ample time to amass tactical intelligence and the resulting TSA process was developed slowly. Commanders and their troops came to know and judge their opponents quite well. With experience, the weather and terrain conditions affecting tactical mobility and performance could be accurately predicted. These conditions essentially remained constant until technology developed a machine capable of rapidly and safely traversing enemy front lines: the tank. Tanks crushed the immutability of the Front and forever changed the methodology of warfare. Rapidly moving tanks and forces demanded that the TSA process be performed more quickly than previously. Today, the ICBM has placed virtually every city in the world within twenty minutes of total destruction. We live in a global village where our units of time continue to shrink with each technological advance. TSA processing must now be performed in minutes rather than days.

Secondly, technology has increased the penalty for TSA error. We live in a time of Strategic Deterrence where the threat of global confrontation is too horrible to contemplate. However, for those with nothing to lose, the threat of Nuclear War is more rhetoric than fact. With stakes this high, accurate and timely vigilance has become a necessity. In tactical situations, TSA error can foster a deadly consequence.

Thirdly, technology has increased our dependence upon automation. For both of the reasons cited above: the rapid shrinkage of situational analysis reaction time and the (potentially) enormous penalty paid for analysis error, the Intelligence Officer and his Staff require assistance. In addition, in Army 21 [Army 21], the U.S. Army predicts that future battles will be fought in more of a guerrilla or skirmish-like fashion. Battles will likely be loosely coordinated with more responsibility given to lower echelon commanders. The Army Intelligence Officer must be prepared to rapidly perform a TSA in a foreign, unfamiliar part of the world, often against an unknown or unfamiliar enemy commander. Curiously, technology appears to offer the only substantive assistance to the Army Intelligence Officer.

II. AUTOMATED TACTICAL SITUATION ASSESSMENT

Technology offers assistance to an Intelligence Officer in the form of computer automation. The promise of automation is found in the computer's ability to rapidly process data. With speed comes an apparent expansion of time, which is exactly what an Intelligence Officer needs: more time to analyze the tactical data and more time to react to an enemy threat.

A. Plans

Basically, the computer can assist an Intelligence Officer if it is able to perform the following tasks: accurately recognize an enemy tactical plan; estimate the probable evolution of the tactical plan; and determine the probable threat of the evolved tactical plan to friendly forces. The key element of this automated process is the recognition of a tactical plan. A tactical plan [Wilensky, 1983; Hatfield; Chubb] is defined as a purposeful, rational sequence of actions which, when executed by some tactical force(s), results in a tactical objective or goal, G, being realized. A plan, P, is represented as a time-contiguous sequence of tuples each of which is executed by a plan actor, AC. That is,

$$P = \{ \{A, CS\}_1, \dots, \{A, CS\}_n \} \quad (2.1)$$

where A_i is some action performed by AC,
and CS_i is the context state which AC
believes, is necessary to initiate execution
of A_i ,
for $i = 1, \dots, n$.

For purposes of this discussion and unless otherwise specified, the plan actor, AC, is assumed to be human. Every plan tuple $\{A, CS\}_i \in P$ contains two related elements: an action, A_i , and an associated context state,

CS_i . Each CS_i is a description of context features or conditions which the actor, AC, believes must be present within the domain to successfully initiate the execution of A_i [Chubb]. A plan P is executed by an actor, AC, by the actor sequentially executing plan tuple actions. AC begins by executing A_1 within context state CS_1 . At the completion of this action, a new (possibly different) context state, CS_2 , results from AC executing A_1 in CS_1 . Action A_2 is now executed within CS_2 . This process continues (if the plan is successful) until the last plan tuple results in a context state expression which contains some plan goal state, G. That is,

$$G_P = AC(P) \quad (2.2)$$

where G_P is read to mean "the goal G of some plan P".

Familiar examples of plans are found in everyday life. The sequence of actions which we use in the morning as we prepare for the day's work represent the tuple elements of a plan called "Get ready for work". Although most plans are performed unconsciously and by rote, plan tuple execution is based upon two Axioms.

If P is a plan where $P = \{ \{A, CS\}_1, \dots, \{A, CS\}_n \}$, then for every $i, i = 1, \dots, n$

Plan Execution Axiom 1: Every tuple $\{A, CS\}_i \in P$ is causally and temporally related to its (i-1) and (i+1) tuple neighbors, and,

Plan Execution Axiom 2: Plan tuple subelements, A_i and CS_i , are variables which are dynamically developed by AC. (2.3)

Contiguous plan tuples are causally related since the completion of the execution of the i th action, A_i , is intended to result in context state CS_{i+1} . If the context state which results from the execution of A_i in CS_i by AC does not equal the predicted CS_{i+1} , then the respective values of A_i , A_{i+1} , and/or predicted context state CS_{i+1} may be dynamically changed by AC. This (human) ability to dynamically change or alter a plan in accordance with contextual changes/queues accounts for the richness of the plan expression and its endless variety. Some plan interactions, however, are so well understood that the plan actor is able to accurately predict the interaction of each tuple within the entire plan.

Definition 2-1: A plan P is defined to be deterministic if and only if for every plan tuple $\{A, CS\}_i \in P, i = 1, \dots, n$, rank $E\{A, CS\}_i = 1$.

The notation used to describe plan tuple execution with its resulting tuple-to-tuple interaction is:

$$E\{A, CS\}_i \rightarrow CS_{i+1} \quad (2.4)$$

where the function E is read to mean " A_i executed within context state CS_i yields the singleton tuple (anticipating the A_{i+1} action) context state expression CS_{i+1} ". As a singleton tuple, $CS_{i+1} = \{CS_{i+1}\}$ and we sometimes write $E\{A, CS\}_i = \{CS_{i+1}\}$ which is equivalent to (2.4).

For any successful plan P , $E\{A, CS\}_n \rightarrow CS_{n+1}$ where $G_P \in CS_{n+1}$ and $\{A, CS\}_n$ is assumed to be the last tuple element in P .

The plan tuple-to-tuple relationship in (2.4) may also be represented using a directed graph *tree* representation [Nilsson]. Each tuple is represented by a *node* while the $E\{A, CS\}_i \rightarrow CS_{i+1}$ relationship is represented by *branches* or *arcs* between nodes. The function E develops context state possibilities resulting from the execution of A_i by AC in context state CS_i . The *branches* represent the causal and temporal flow from plan tuple to tuple. The tree representation of $E\{A, CS\}_i$ for every tuple in P is called an *expanded form* tree representation, P_{EF} , of plan P . P_{EF} represents every possible action and context state relationship in one representation for one plan (Figure One).

We define M to be the number of nodes in P_{EF} . For a deterministic plan, $M_{deterministic} = n$, where n is the number of tuples in plan P . However, not every plan is deterministic. Some plans contain tuples such that $E\{A, CS\}_i$ is not single-valued. For example, consider the $A_i = \text{"flip a coin"}$. The context state expression resulting from $E\{A, CS\}_i$ may contain two possible states: heads or tails. The number of possible contextual state results for $E\{A, CS\}_i$ is, however, *finite*.

Definition 2-2: A plan P is defined to be *N-deterministic* if and only if for every plan tuple in P , $i = 1, \dots, n$, $\text{rank } E\{A, CS\}_i \leq N$ where $N > 1$ and finite and there exists at least one plan tuple $\{A, CS\}_i \in P$ such that $\text{rank } E\{A, CS\}_i = N$.

Notation used to indicate that the *rank* of the expected value of an i th plan tuple is greater than one is:

$$E_j\{A, CS\}_i \rightarrow \{CS_u, \dots, CS_{u+j-1}\} \text{ for all } j > 1.$$

The execution function E_j here produces a set of context state tuples which could also be written as,

$$E_j\{A, CS\}_i \rightarrow \{\{CS_u\}, \dots, \{CS_{u+j-1}\}\}.$$

A least upper bound for the value of $M_{N-deterministic}$ is,

$$1 + \sum_{i=2}^n [\max [\text{rank } E\{A, CS\}_i] \times \max [\text{rank } E\{A, CS\}_{i-1}]] \quad (2.5)$$

where n is the number of tuples in P .

Plans which are neither deterministic nor N -deterministic are defined to be *nondeterministic*.

Definition 2-3: A plan P is defined to be *nondeterministic* if and only if there exists some plan tuple, $\{A, CS\}_i \in P$ such that $\text{rank } E\{A, CS\}_i$ is not finite. Notation used to indicate a countable number of possibilities is \bar{E} or, $\bar{E}\{A, CS\}_i \rightarrow \{CS_2, \dots\}$.

Nondeterministic plans usually include actions and context states which are best described as random or probabilistic with countably many possibilities. The magnitude of $M_{nondeterministic}$ is, by definition,

infinitely large. Fortunately, most tactical plans of interest are N -deterministic. The magnitude of $M_{N-deterministic}$, however, can be far too large to permit practical computer implementation of the plan.

III. COMPUTER MODELS AND LEVELS OF ABSTRACTION

For automation to be an effective plan recognition device, one must be able to describe a tactical plan in sufficient detail. It is quite easy to describe most deterministic plans. Powerful computer programs exist which are able to accurately recognize (and create) some N -deterministic plans. For example, Samuel's checker playing program is able to play checkers at least as well as any human. The reason for this is that the size of the expanded form representation, $M_{checkers}$, for this computer plan is small enough to be fully expressible within computer memory. As a result, Samuel's program can "look ahead" in the expanded form tree representation and examine alternative context-states, given some checkers move action (i.e., $E_j\{A, CS\}_i$). The game becomes fully deterministic and the best move is always made. The program never loses; at worst, the result is a draw.

The game of chess, on the other hand, is an example of an N -deterministic plan where the magnitude of M_{chess} is far too large to be fully represented within computer memory. Recent cognitive research [Katter] suggests that although humans lack the ability to accurately recall the details of very large M valued plans, they are able to rapidly recognize these plans. These findings have suggested the following:

Plan Abstraction Axiom: Plans with large $M_{expanded\ form}$ values can be faithfully represented, through a process of plan abstraction, in a plan form such that $M_{abstracted} \ll M_{expanded\ form}$.

This axiom assumes the existence of some type of transformation which maps (abstracts or reduces) the expanded form representation of P into an abstracted and implementable plan form.

A. Plan Abstraction

We begin with the P_{EF} representation of an N -deterministic plan P (Figure One). By definition, every tree node has at most a single tuple successor called its *Father* node. That is, if $\{A, CS\}_j \in P_{EF}$, then there exists a *Father* function such that,

$$\begin{aligned} \text{either } & \text{Father}\{A, CS\}_j = \{A, CS\}_f \in P_{EF}, \\ \text{or } & \text{Father}\{A, CS\}_j = \emptyset. \end{aligned} \quad (3.1)$$

where $\{A, CS\}_f \neq \{A, CS\}_j$, $\{A, CS\}_f \in P_{EF}$, and $\{A, CS\}_j \in E_j\{A, CS\}_f$.

The P_{EF} tree root node is represented by the first plan tuple in P , $\{A, CS\}_1$. By definition, tuple $\{A, CS\}_1$ is the only node in P_{EF} such that $\text{Father}\{A, CS\}_1 = \emptyset$.

Without loss of generality, we also assume that the plan P has concluded successfully and that there exists some non-empty subset of terminal nodes $\subset P_{EF}$ which contain the desired state, G_P . We will only examine successful plan strategies by recursively removing all of the terminal nodes in P_{EF} which do not contain G_P . Without loss of generality, P_{EF} may be represented as the set of remaining successful plans. That is, each tree terminal node in P_{EF} represents a plan tuple for a successfully concluded variation of plan P .

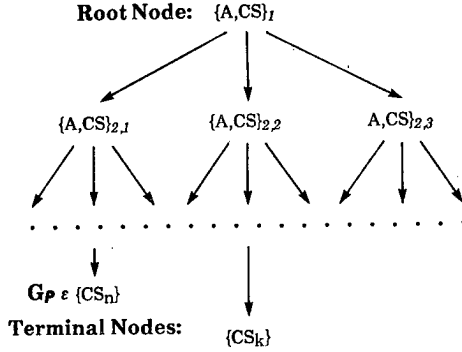


Figure One: Expanded Form P_{EF} of plan P

Definition 3-1: Define the set $W_j \subset P_{EF}$ to be:
for every terminal node $\{A, CS\}_j \in P_{EF}$, let
 $W_j = \{ \{A, CS\}_j \cup \{ \{A, CS\}_g : \{A, CS\}_g \in P_{EF} \text{ and } \{A, CS\}_g = \text{Father} \{A, CS\}_j \text{ where } \{A, CS\}_j \in W_j \}$.

Let Y be the set of all W_j as defined. We assume that Y is non-empty with $\text{rank } Y = p > 0$. The following Lemma may be easily proved using Definition 3-1.

LEMMA 3.1: Given the W_j as defined in 3-1. Then,

1. Each W_j is a plan.
2. The first tuple in each W_j is the root node of P_{EF} .
Conversely, the last tuple in each W_j is the j th terminal node in P_{EF} .
3. $\cup W_j = P_{EF}$
4. $\cap W_j \neq \emptyset$
5. Every W_j is well-ordered.

Our examination of plan abstraction will proceed as follows. First, we will restrict our study domain to the elements of Y . We will examine the set of plan abstraction transformations which map tuples from W_j and W_k to a single plan tuple. Next, we will examine the set of plan abstraction transformations which map tuples from a single W_j to itself. Finally, we will show that these plan abstraction transformations may be used to reduce (abstract) appropriate subsets of P_{EF} .

We begin by describing two relations, R_G and R_T on the set S of all plans executed by actor AC .

Definition 3-2: Let B be a partition of S . Define the relation $R_G = \{ (x, y) : x \text{ and } y \text{ are plans in } S; \text{ there exists a class } C \in B \text{ such that } x, y \in C \text{ if and only if } x \text{ and } y \text{ share the same actor goal, } G_j \}$.

LEMMA 3.2: R_G is an equivalence relation on S .

Proof: Obviously R_G is reflexive (xR_Gx) and symmetric (if xR_Gy , then yR_Gx). Assume xR_Gy and yR_Gz then, by definition, there exists classes $C, C' \in B$ with $x, y \in C$ and $y, z \in C'$. Since $y \in C \cap C'$ the classes are not disjoint, therefore $C = C'$. Therefore, $x, z \in C \rightarrow xR_Gz$. Therefore R_G is transitive. q.e.d.

For every partitioned class C resulting from R_G there exists $P_{EF} \in C$ and set $Y = \{W_j : W_j \text{ a successful plan in class } C\}$.

Definition 3-3: Assuming that Y is a non-empty set, and B be a partition of Y . Define the relation $R_T = \{ (x, y) : x \text{ and } y \in Y; \text{ there exists a class } C \in B \text{ such that } x, y \in C \text{ if and only if } x \text{ and } y \text{ have equivalent plan execution times} \}$.

It can be easily shown that,

LEMMA 3.3: R_T is an equivalence relation on Y .

Unless otherwise noted, we will assume that any discussions involving two or more W_j tacitly assume that the plans are members of the same class defined by the relation $\{ (x, y) : xR_Gy \text{ and } xR_Ty \}$. That is, for all x, y under consideration, x and y

- a. belong to the actor, AC
- b. share the same actor goal
- c. are successful plans, and
- d. have equivalent plan execution times.

IV. PLAN ABSTRACTION TRANSFORMATIONS

Definition 4-1: Let P be a plan with actor AC , initial context state CS_1 , and plan goal G_P . Assume F is a non-empty subset of P and F a plan. Then T_A is a plan abstraction transformation of F if $T_A: F \rightarrow F'$ where F' a plan, $\text{rank } F' < \text{rank } F$, and if F' replaces F in P , $AC(P) \rightarrow G_P$.

A. Abstracting Tuples from Two Deterministic Plans

We begin by enumerating the types of plans which can be abstracted (reduced) to plans of lesser rank. We enumerate the possible tuple-to-tuple relationships which may be formed from tuples which belong to two different plans, W_j and W_g , where tuple $\{A, CS\}_j \in W_j$, $\{A, CS\}_g \in W_g$, and $W_j \neq W_g$. Plan tuple variables to be considered are:

- a) Actions: A_j and A_g
- b) Initial Context States: CS_j and CS_g ,
- c) Final Context States: $CS_{j+1} \in E\{A, CS\}_j$,
 $CS_{g+1} \in E\{A, CS\}_g$.

For each of the above categories (Actions, Initial CS, Final CS), the tuple variables are either *equal* or *not equal*. Hence, there are 23 possible tuple state descriptions. We will prove that some of these tuple state descriptions can be abstracted (a T_A exists) and that the remaining state descriptions can not be abstracted.

For each of the possible tuple state descriptions we assume that W_j, W_g are deterministic plans. This assumption is, however, not necessarily true. By construction, each of the W_j appears to be deterministic. However, given any two contiguous tuples $\{A, CS\}_i, \{A, CS\}_{i+1} \in W_j$, either:

1. $E\{A, CS\}_i \rightarrow CS_{i+1} \in \{A, CS\}_{i+1}$
(deterministic relationship)
2. $E_j \{A, CS\}_i \rightarrow \{CS_{i+1}, \dots, CS_j\}$.
(N-deterministic relationship)

THEOREM H1: *Given the following tuple state description ($A_j = A_g; CS_j = CS_g; CS_{j+1} = CS_{g+1}$), then there exists tuple $\{A, CS\}_x$ and abstraction transformation $TH1$ such that,*
 $TH1 : \{\{A, CS\}_j, \{A, CS\}_g\} \rightarrow \{\{A, CS\}_x\}.$

Proof: By definition, $\{A, CS\}_j = \{A, CS\}_g$. Clearly $\{\{A, CS\}_j, \{A, CS\}_g\}$ is an N-deterministic plan with input context states equal to $E_2\{A, CS\}_k = \{\{CS_j\}, \{CS_g\}\}$ for some N-deterministic tuple $\{A, CS\}_k$ and with deterministic plan tuple possibilities $\{A, CS\}_j$ and $\{A, CS\}_g$. Let $\{A, CS\}_x = \{A, CS\}_j$ where $\{\{A, CS\}_j\}$ a plan by definition. Then,
 $rank\{\{A, CS\}_j\} + rank\{\{A, CS\}_g\} > rank\{\{A, CS\}_x\}$ and
 $E\{A, CS\}_x = E\{A, CS\}_j = E\{A, CS\}_g$ since $CS_{j+1} = CS_{g+1}$. Therefore $TH1$ is the required abstraction transformation.

THEOREM H2: *Given the following tuple state description ($A_j = A_g; CS_j \neq CS_g; CS_{j+1} = CS_{g+1}$), then there exists tuple $\{A, CS\}_x$ and abstraction transformation $TH2$ such that,*
 $TH2 : \{\{A, CS\}_j, \{A, CS\}_g\} \rightarrow \{\{A, CS\}_x\}.$

Proof: $\{\{A, CS\}_j, \{A, CS\}_g\}$ is an N-deterministic plan with input context states equal to $E_2\{A, CS\}_k = \{\{CS_j\}, \{CS_g\}\}$ for some N-deterministic tuple $\{A, CS\}_k$ and with deterministic plan tuple possibilities $\{A, CS\}_j$ and $\{A, CS\}_g$. Since $E\{A, CS\}_j = E\{A_j, CS_g\} = E\{A_g, CS_j\}$, the actor action is independent of value of $E_2\{A, CS\}_k$. Therefore, let $\{A, CS\}_x = \{A, CS\}_j$ where $\{\{A, CS\}_j\}$ a plan by definition. $Rank\{\{A, CS\}_j\} + rank\{\{A, CS\}_g\} > rank\{\{A, CS\}_x\}$ and $E\{A, CS\}_x = E\{A, CS\}_j = E\{A, CS\}_g$ since $CS_{j+1} = CS_{g+1}$, as required. Therefore $TH2$ is the required abstraction transformation.

THEOREM H3: *Given the following state description ($A_j \neq A_g; CS_j = CS_g; CS_{j+1} = CS_{g+1}$), then there exists tuple $\{A, CS\}_x$ and abstraction transformation $TH3$ such that, $TH3 : \{\{A, CS\}_j, \{A, CS\}_g\} \rightarrow \{\{A, CS\}_x\}.$*

Proof: $\{\{A, CS\}_j, \{A, CS\}_g\}$ is an N-deterministic plan with input context states equal to $E_2\{A, CS\}_k = \{\{CS_j\}, \{CS_g\}\}$ for

some N-deterministic tuple $\{A, CS\}_k$ and with deterministic plan tuple possibilities $\{A, CS\}_j$ and $\{A, CS\}_g$. Since $E\{A_j, CS_g\} = E\{A_g, CS_j\}$ and $A_j \neq A_g$, action differences do not effect the executed tuple values, i.e., $CS_{j+1} = CS_{g+1}$. Therefore, let $\{A, CS\}_x = \{A, CS\}_j$ with $\{\{A, CS\}_j\}$ a plan by definition. Clearly,
 $rank\{\{A, CS\}_j\} + rank\{\{A, CS\}_g\} > rank\{\{A, CS\}_x\}$ and
 $E\{A, CS\}_x = E\{A, CS\}_j = E\{A, CS\}_g$ as required. Therefore $TH3$ is the required abstraction transformation.

THEOREM H4: *For each of the following deterministic state descriptions there exists no abstraction transformation T_A and plan tuple $\{A, CS\}_x$ such that*
 $T_A : \{\{A, CS\}_j, \{A, CS\}_g\} \rightarrow \{\{A, CS\}_x\}.$

- a. ($A_j \neq A_g; CS_j \neq CS_g; CS_{j+1} \neq CS_{g+1}$)
- b. ($A_j \neq A_g; CS_j = CS_g; CS_{j+1} \neq CS_{g+1}$)
- c. ($A_j = A_g; CS_j \neq CS_g; CS_{j+1} \neq CS_{g+1}$)
- d. ($A_j = A_g; CS_j = CS_g; CS_{j+1} \neq CS_{g+1}$)
- e. ($A_j \neq A_g; CS_j \neq CS_g; CS_{j+1} = CS_{g+1}$).

Proof For State Descriptions a, b, c, d: In each of these descriptions $E\{A, CS\}_j \neq E\{A, CS\}_g$ since $CS_{j+1} \neq CS_{g+1}$. Since $\{\{A, CS\}_x\}$ a deterministic plan, $E\{A, CS\}_x \neq \{CS_{j+1}, CS_{g+1}\}$ and therefore the required T_A does not exist.

Proof For State Description e: In this description $E\{A, CS\}_j = E\{A, CS\}_g$ since $CS_{j+1} = CS_{g+1}$. Let $\{\{A, CS\}_j, \{A, CS\}_g\}$ be an N-deterministic plan with input context states equal to $E_2\{A, CS\}_k = \{\{CS_j\}, \{CS_g\}\}$ for some N-deterministic tuple $\{A, CS\}_k$ and with deterministic plan tuple possibilities $\{A, CS\}_j$ and $\{A, CS\}_g$. However, since $A_j \neq A_g$ and $CS_j \neq CS_g$, there exists no CS_z such that $\{A_j, CS_z\} = \{A_g, CS_z\}$ or an A_z such that $\{A_z, CS_g\} = \{A_z, CS_j\}$. Therefore the required $\{\{A, CS\}_z\}$ does not exist, and therefore the required T_A does not exist.

COROLLARY 4.1: *Given any two deterministic plans, W_j, W_g , with tuples $\{A, CS\}_j \in W_j$ and $\{A, CS\}_g \in W_g$, then there exists an abstraction transformation $TH4$ and plan tuple $\{A, CS\}_x$ such that*

$TH4 : \{\{A, CS\}_j, \{A, CS\}_g\} \rightarrow \{\{A, CS\}_x\}$ if and only if $E\{A, CS\}_j = E\{A, CS\}_g$ and $A_j = A_g$ or $CS_j = CS_g$.

Proof: Follows directly from Theorems H1-4.

B. Abstracting Tuples From Two N-deterministic Plans

We now consider the plan domain of N-deterministic plans, K_j and $K_g \subset P_{EF}$. We are assured, without a loss of generality, that such K_i can be found since P_{EF} N-deterministic. We now examine those conditions under which plan tuples from N-deterministic plans may be abstracted to a single N-deterministic plan tuple.

THEOREM H5: *If K_j and K_g are N-deterministic plans with tuples $\{A, CS\}_j \in K_j$ and $\{A, CS\}_g \in K_g$, such that $E_5\{A, CS\}_j = E_5\{A, CS\}_g$ and $A_j = A_g$ or $CS_j = CS_g$, then there exists a plan tuple $\{A, CS\}_x$ and*

abstraction transformation **TH5** such that,
TH5 : $\{\{A, CS\}_j, \{A, CS\}_g\} \rightarrow \{\{A, CS\}_x\}$.

Proof: Since the only difference between deterministic and N-deterministic plan tuples is the value of the tuple executed value, we assert the following:

1. If $\{A, CS\}_j = \{A, CS\}_g$ then, by Theorem H1, let $\{\{A, CS\}_x\} = \{\{A, CS\}_j\}$.
2. If $CS_j \neq CS_g$ then, by Theorem H2, let $\{\{A, CS\}_x\} = \{\{A, CS\}_j\}$.
3. If $CS_j = CS_g$, then by Theorem H3, let $\{\{A, CS\}_x\} = \{\{A, CS\}_j\}$.

Clearly, for each of the above state descriptions $\{\{A, CS\}_x\}$ is an N-deterministic plan where, $\text{rank } \{\{A, CS\}_j\} + \text{rank } \{\{A, CS\}_g\} > \text{rank } \{\{A, CS\}_x\}$ as required. Likewise, by definition, $E_S\{A, CS\}_j = E_S\{A, CS\}_g = E_S\{A, CS\}_x$ as required. Therefore, **TH5** is the required abstraction transformation.

C. Abstracting Tuples From One Deterministic Plan

We now consider the special case of abstracting tuples from a single deterministic plan, W_j . That is, if $\{A, CS\}_j$ and $\{A, CS\}_{j+1}$ are any two contiguous plan tuples belonging to some deterministic plan, W_j , then does there exist a deterministic plan tuple $\{A, CS\}_x$ and an abstraction transformation **TA** such that,
TA : $\{\{A, CS\}_j, \{A, CS\}_{j+1}\} \rightarrow \{\{A, CS\}_x\}$? The plan tuple variables being considered (Figure Two) are:

- a) Actions A_j and A_{j+1}
- b) Initial Context States CS_j and CS_{j+1}
- c) Final Context State: $CS_{j+2} = E\{A, CS\}_{j+1}$.

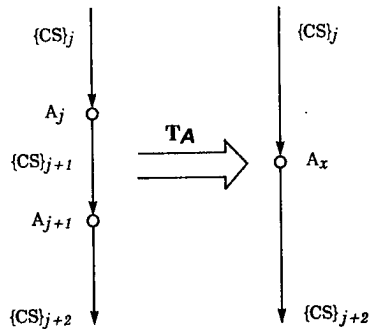


Figure Two: Abstracting a Single Deterministic Plan.

THEOREM V1: If W_j is a deterministic plan with contiguous plan tuples $\{A, CS\}_j$ and $\{A, CS\}_{j+1} \in W_j$, then there exists a deterministic plan tuple $\{A, CS\}_x$ and an abstraction transformation **TV1** such that **TV1** : $\{\{A, CS\}_j, \{A, CS\}_{j+1}\} \rightarrow \{\{A, CS\}_x\}$ if and only if $CS_j = CS_{j+1}$.

Proof: The conditions under which $\{A, CS\}_j$ and $\{A, CS\}_{j+1}$ may be abstracted by some **TV1** into a single deterministic plan tuple, $\{A, CS\}_x$ are those which satisfy the following equation,

$$E(A_x, CS_j) \triangleq E\{A_{j+1}, E\{A_j, CS_j\}\}. \quad (4.1)$$

Since W_j is a deterministic plan, (3.1) may be rewritten as, $E(A_x, CS_j) = E\{A_{j+1}, E\{A_j, CS_j\}\}$. Therefore $A_x = A_{j+1}$ and $CS_j = E\{A_j, CS_j\} = CS_{j+1}$, and $\{A, CS\}_x = \{A, CS\}_{j+1}$. Conversely, $CS_j = CS_{j+1} = E\{A_j, CS_j\}$ by (4.1). Therefore, $A_x = A_{j+1}$ and $\{A, CS\}_x = \{A, CS\}_{j+1}$. In both cases, $\text{rank } \{\{A, CS\}_x\} < \text{rank } \{\{A, CS\}_{j+1}\} + \text{rank } \{\{A, CS\}_j\}$. Therefore, **TV1** is the required transformation,
TV1 : $\{\{A, CS\}_j, \{A, CS\}_{j+1}\} \rightarrow \{\{A, CS\}_x\}$. q.e.d.

Comment: Given some $(j+1)$ th tuple in a deterministic plan W_j , it is possible to abstract the j th and $(j+1)$ th plan tuples if and only if $E\{A_j, CS_j\} \rightarrow CS_j$. That is, action A_j , acting in context state CS_j , does not alter conditions/features in CS_j , e.g., the A_x acts like the identity function with respect to CS_j with the context state fixed for "j" tuples. In this case the CS_x essentially acts as a "contextual stage" for the actor actions. Examples include domain independent actions.

Theorem V1 can be applied repeatedly to some deterministic plan P with $\text{rank } P > 2$. That is, if

$$P = \{\{A, CS\}_1, \dots, \{A, CS\}_v\},$$

then P can be recursively abstracted to form a new deterministic plan, P' , where,

$$P' = \{\{A, CS\}_v\}$$

and the A_v represents a composite description for (plan P) actions $\{A_1, \dots, A_v\}$.

D. Abstracting Tuples from One N-deterministic Plan

We now consider the case where K_j is a N-deterministic plan, $K_j \subset P_{EF}$.

THEOREM V2: If $\{A, CS\}_j, \{A, CS\}_{j+1}$ are any two contiguous plan tuples belonging to N-deterministic plan K_j , such that $E_f\{A, CS\}_j$ for $f > 1$ and $E_h\{A, CS\}_{j+1}$ for $h > 1$, then there exists no abstraction transformation **TV2** and N-deterministic plan tuple $\{A, CS\}_x$ such that,

$$\text{TV2} : \{\{A, CS\}_j, \{A, CS\}_{j+1}\} \rightarrow \{\{A, CS\}_x\}.$$

Proof: As in Theorem V1, we consider those conditions such that equation (4.1) is true (Figure Three). In particular,

$$CS_j \triangleq E\{A, CS\}_j$$

is never true since, $CS_j \neq \{CS_q, \dots, CS_{q+f-1}\} = E\{A, CS\}_j$ where by definition $f > 1$ since $\{A, CS\}_j$ an N-deterministic plan tuple in K_j . q.e.d.

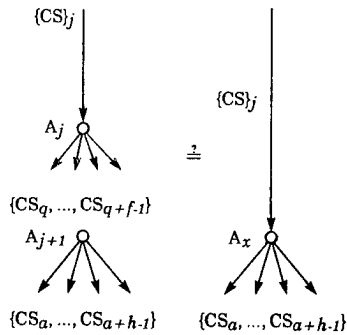


Figure Three: Abstracting a single N-deterministic Plan.

A simple corollary to Theorem V2, which is not proved, is,

COROLLARY V2a: *If $\{A, CS\}_j, \{A, CS\}_{j+1}$ are any two contiguous plan tuples belonging to N-deterministic plan K_j , such that $\text{rank } E\{A, CS\}_j = 1$, then there exists an abstraction transformation TV_{2a} and N-deterministic plan tuple $\{A, CS\}_x$ such that,*
 $TV_{2a} : \{\{A, CS\}_j, \{A, CS\}_{j+1}\} \rightarrow \{\{A, CS\}_x\}.$

V. PLAN ABSTRACTION TRANSFORMATION SUMMARY

We have shown that *deterministic plans* may be accurately abstracted using the following transformations: TV_1 (see Theorem V1), and TH_4 (see Corollary 4.1), as applicable. Both of these abstraction transformations, however, require extensive a priori information concerning the plan's,

- A1. **Context State**, when using transformation TV_1 , or,
- B1. Either **Context State** or **Actor Action**, when using transformations TH_4 .

In particular, TV_1 may be employed for plan actions which are insensitive to elements/features found within the actor's contextual domain, CS.

N-deterministic plans, however, are only abstractable whenever,

- A2. A neighboring plan tuple appears deterministic, or,
- B2. **Tuple Execution values are equal**, and known a priori.

If case A2 above is true, transformation TV_{2a} may be used. TH_5 may be employed when case B2 conditions are true. In general, N-deterministic plans which contain few deterministic plan tuple elements are *not* good candidates for plan abstraction.

VI. PLAN ABSTRACTION ERRORS

The TASC TSA research effort has been directed, under U.S. Government contract, to develop a *domain independent* and *extensible* TSA paradigm. TSA depends upon the system's ability to recognize ongoing or developing enemy tactical plans. The TASC TSA automated *plan recognizer paradigm* is the problem solver which attempts to recognize emerging or ongoing tactical events by comparing intelligence sensor data with a set of instantiated tactical scripts/plans. Incoming sensor and terrain data are first analyzed by the TSA *Plan Design Module (PDM)*. Using the terrain information and a knowledge of the principles of warfare, the PDM produces output that helps parameterize the skeleton tactical scripts which are contained in a script library (scripts written by The Analytic Sciences Corporation HERO group of military experts and historians). The purpose of the *Plan Evaluation Module (PEM)* is to associate the preconditioned tactical scripts with processed intelligence data. Temporal and causal relationships between tactical scripts are represented using a somewhat modified [Allen, 1983, 1985] version of Wilensky's *KODIAK* plan relation language [Wilensky, 1984]. Scripts are used in *KODIAK* to represent highly stereotypical tactical information which corresponds to generally accepted practices for conducting warfare.

KODIAK is a relation-based representation language where the objects of the relation(s) are **absolutes**, **relations** and **aspectuals** [Hatfield]. *Absolutes* can represent physical objects (AC, or CS_i), actions (A_i), events (CS_i) or abstract ideas (G_p). *Relations* capture relations between absolutes and *aspectuals* are the arguments of the relation. Temporal and causal relationships are captured using *associations* between Objects. Figure Four shows a *KODIAK* representation of a Maneuver Plan [Hatfield]. A *KODIAK* representation of an enemy tactic resembles an augmented N-deterministic plan of tactical scripts. The plan is augmented in the sense that both causal and temporal relations between objects are specified. *KODIAK* scripts may also be represented as plans. We will now prove that if these tactical scripts are *improperly abstracted*, then the PEM will *not* function properly.

A. KODIAK Action Scripts

Each *KODIAK* Action generally contains several slots which represent action *preconditions* and *postconditions*. *KODIAK* actions may also be represented as a plan where *action names* represent *plan names* or symbols, *action preconditions* represent features in CS_1 believed to be necessary for the plan to be executed, and *action postcondition(s)* represent plan goals or goal context states. For example, the *KODIAK* Action "red breakthrough at loc 1" is the name of a plan whose goal is "red controls loc 1". The important *contextual features* to check which make the execution of plan *Red Breakthrough at loc 1* possible are given as action preconditions "Blue Gap at loc 1" and "Red Force at loc 1" (Figure Four). That is,

Red Breakthrough at loc 1 =
 $\{[A, CS]_1, \dots, [A, CS]_k\} \rightarrow \text{Red Controls loc 1}$ (6-1)

where **blue gap at loc 1** and **red force at loc 1** are states/conditions/features which must be present within CS_1 .

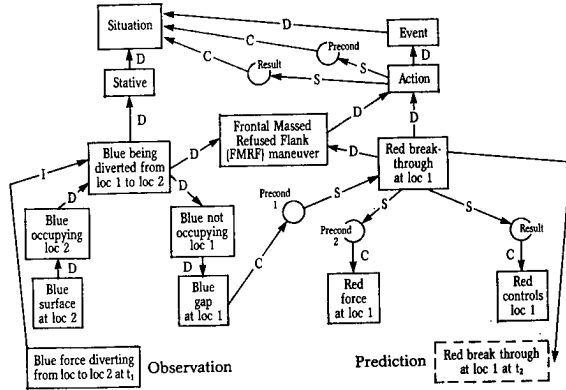


Figure Four: Representation of Frontal Assault With Massed and Refused Flanks.

Other than *action preconditions and postconditions*, no other *checks (conditions/states)* are used to check the (ongoing) status of the KODIAK action (see Section B and *Named Areas of Interest*). We now prove that this representation of tactical script-based action is only correct when the (action) plan is either *deterministic* or can be represented as the *union of a finite number of very simple deterministic plans*.

THEOREM 6-1: Let P be a plan with a priori specified context CS_1 and plan goal context CS_{n+1} such that $G_P \in CS_{n+1} = E\{A, CS\}_n$.
 If $P' = \{[A, CS]_i : \forall i, 1 < i < n, \text{ where } [A, CS]_i \in P \text{ and } AC(P) = G_P\}$, where $P' \neq \emptyset$, then P' is a deterministic plan.

Proof: Assume $P = \{[A, CS]_1, \dots, [A, CS]_n\}$ with $E\{A, CS\}_n \rightarrow CS_{n+1}$ as specified. P can be written as $P = \{[A, CS]_1, P', [A, CS]_n\}$ where $P' \subset P$ and P' a plan. Now assume for any tuple $[A, CS]_e \in P'$, $E\{A_e, CS_e\} = E\{A_e, CS_x\}$ for any CS_x , in particular for $CS_e \neq CS_x$. Show that P' deterministic.

Since $E\{A_e, CS_e\} = E\{A_e, CS_x\}$, then $E\{A_e, CS_e\} = E\{A_e, E\{A, CS\}_{e-1}\}$ for all $[A, CS]_h$, $h=1, \dots, (e-1)$. In particular, then $E\{A_e, CS_e\} = E\{A_e, E\{A, CS\}_1\}$ for $[A_e, CS_e]$ the last tuple in P' . Then $P = \{[A, CS]_1, [A, CS]_e, [A, CS]_n\}$ where $P' = \{[A, CS]_e\}$. If tuple $[A, CS]_1$ deterministic, then $\{[A, CS]_1, [A, CS]_e\}$ deterministic, and therefore, P' deterministic, as required. Assume $\{A, CS\}_1$ N-deterministic, where $E\{A_e, CS_1\} = \{CS_{i_1}, \dots, CS_{p_1}\}$. Since $E\{A_e, CS_i\} = E\{A_e, CS_{p_1}\}$ for every element of $E\{A_e, CS_1\}$, and A_e equal for all such tuples, by Corollary 4.1 these $\{A_e, CS_i\}$ can be abstracted to a single tuple. That is, $\{[A, CS]_1, [A, CS]_e\}$ may be written as some $\{[A_e, CS_1]\}$. Therefore, P' deterministic, as required. q.e.d.

COROLLARY 6.1: If plan P as in Theorem 6-1, then either P deterministic, or there exists n deterministic plans W_j such that,

$$P = \bigcup_{j=1}^n W_j.$$

Each W_j , if it exists, has a very simple deterministic form equal to $\{[A_e, CS_1], [A, CS]_2\}$ where, $E\{A, CS\}_2 = \{CS_{n+1}\}$ as required.

B. Plan Projection/Evaluation Errors Resulting from Improperly Represented Action Plans

If a KODIAK *action script* doesn't have a plan form as described in Corollary 6.1, then, at a minimum, subtle errors will occur within the PEM process. If the action plan form is incorrect, the PEF form of the plan can be quite large and complicated with numerous (unforeseen) action/contextual state interactions with other features within the tactical domain, e.g., other forces, environment. The KODIAK methodology of only checking prescribed action preconditions and post conditions will be insufficient to accurately monitor the script action. The action script, improperly represented, will not represent stereotypical but rather domain dependent action(s), and the subsequent execution of the action will depend heavily upon prevailing domain context. Fortunately, these types of PEM errors can be eliminated or significantly reduced by making use of any/all of the following recommendations.

1. Abstract the script properly into a plan form as described in Corollary 6.1.
2. Rewrite the action script into a deterministic form, as described in Corollary 6.1.
3. If the script can't be rewritten into a deterministic form, but is truly N-deterministic, then either,
 - a. rewrite it in a form resembling equation Lemma 3.1, part 3, noting each CS_i and check for each CS_j as part of the KODIAK pre-post condition check, or,
 - b. rewrite the KODIAK representation (e.g., Figure Four) into a form which makes it possible to implement 1, 2, or 3 above.

It has been recently brought to the attention of the author that TASC had forseen this type of problem and had discussed it, albeit briefly, as an implementation issue [Hatfield, p. 12-14] called *Named Areas of Interest (NAI)*. The NAI are computed features (CS_i) which are checked to monitor the progress of ongoing actions. The NAI are similar to recommendation 3a above. NAI's are computed using terrain features (see *Create Plan Network Submodule*). Mr. Stephen Williams, principal investigator for the TASC TSA paradigm implementation effort, has spoken with the author concerning a similar problem. TASC is attempting to develop criteria to evaluate appropriate movement (actions) which appear to be suboptimal locally but which, nevertheless, successfully contribute to an optimal global solution. Work is presently underway between TASC and the author to incorporate some of the theoretics developed within this paper into the PEM.

C. A Situation Analysis Paradox

A situation assessment analysis paradox can arise when using the TASC proposed TSA paradigm with incorrectly abstracted KODIAK action scripts. The PEM will incorrectly assume that the action scripts have a deterministic plan form. As such, the PEM has been programmed to make a closed set of assumptions about plan actions which are no longer necessarily true. For example, assume there exists some KODIAK action plan P with precondition P_{PRE} and post condition P_{POST} which have been defined a priori, as described in section VI-b. Once state P_{POST} is detected by PEM, the action P is assumed to be completed. This conclusion, however, is *only* true when P has the correct plan form (Corollary 6.1) and when both P_{PRE} and P_{POST} have been reliably detected. If P has been incorrectly abstracted, for example, then the presence of state P_{POST} may have *nothing to do* with either the ongoing or completed state of action P. Given this kind of PEM performance, a TSA analysis paradox can arise as follows.

C1. Uniform Probability Distribution Assumption

Assume that for some action P that the likelihood that state P_{POST} is both present in the domain and accurately detected by the PEM is best described as a random process, i.e., *uniform probability distribution*. Then, the likelihood that a plan recognizer could *accurately detect* the presence of some action, P_i , and *discriminate* P_i from some other action, P_j , is problematic. A situation assessment system, relying upon accurate, and robust, plan detection will tend to focus on whatever tactical scenario can be accounted for with the least amount of error. Unfortunately, the plan recognition mechanism is not only used to accurately *detect plans*, but is also used to *prune* plan scenarios from further consideration. A TSA system which is incapable of accurate low-level plan action discrimination will eventually consider a broad base of possible tactical hypotheses. The system will attempt to focus its attention on that tactical script which appears "best". Discernment of a "best" hypothesis involves a prescription-based validation process where the presence or absence of domain features and attributes is used to establish the belief in an ongoing tactical hypothesis. At this point, one of two possible conclusions will be developed. If the low-level discrimination process involves *monitoring tactical KODIAK actions*, then the system will randomly find evidence for/against *every* possibility. If the low-level discrimination process *does not rely upon evidence accrued using KODIAK actions*, then the system will evolve through a (possibly lengthy) process of hypothesis testing where the evidence for previously considered hypotheses will tend to accumulate over time. In either case, the system will eventually arrive at a position where there exists sufficient evidence to posit the presence of a large number of competing tactical scenarios, but with no "clearly best" tactical scenario as yet targeted. Faced with this body of contending tactical hypotheses, each accumulating evidence for its existence in time where each hypothesis is considered equally likely, the system will be forced to make use of a priori tactical threat information.

D. Effect of PEM Errors Upon Tactical Command and Control

When faced with a variety of possible threats, where each is equally likely to represent an opposing enemy tactic, the most conservative rationale is to focus attention on those tactical indicators which reliably establish/discard the presence of enemy tactics which pose the *greatest threat* to friendly forces. This is a type of "worst case scenario" logic which is used when some enemy action appears imminent but unknown. If one is unable to decide what an adversary is planning, but strongly suspects that some adversarial plan is ongoing, then it becomes imperative to thoroughly check for the presence of those adversarial plan(s) which can potentially inflict the most damage. If sufficiently motivated, on the basis of assumed threat, to accurately detect the presence of some plan indicator(s), the PEM as described herein will eventually find evidence for such. The fact that the threat evidence will not be consistently detected is easily rationalized in terms of the importance of this information. Information which is most critical is apt to be that which is most difficult to detect. This type of TSA logic will inevitably result in a system performance converging to one of two modes: either 1) assessing the tactical domain as essentially undecipherable, where every possible enemy tactic is seen as equally likely, or, 2) adopt a "worst case" scenario where the Blue Force Commander's worst fears are realized. Both situation assessments are, of course, in error. Worse than that, however, the former can hide or mask real danger, while the latter can provoke a dangerous Blue Force overreaction which can result in a rapid escalation of the tactical situation.

VII. CONCLUSIONS

A mathematical basis for plan abstraction (reduction) has been developed. We have shown that the plan recognition process, as evidenced by the KODIAK use of tactical, script-based actions, will be inaccurate if the action scripts are not written in a plan form as described in Corollary 6.1 of this paper. Some evidential and experimental evidence for these theoretical predictions has been reported. Recommendations have been made to remove or minimize these errors. However, at least one recommendation would possibly require frequent contextual checks of ongoing action execution. These additional processing requirements may make an automated TSA process less attractive to the user.

We concluded by noting that plan recognition errors are particularly insidious since they may mask other system errors. The result can be a type of analysis paradox where the system is unable to distinguish hypothetical enemy plans from factual ones. Commanders will find it difficult or even dangerous to make command decisions since (given sufficient time) the TSA may converge to an enemy "worst-case" scenario.

REFERENCES

1. J. F. Allen, "Maintaining Knowledge About Temporal Intervals", Comm. of the ACM 26:832-843 (1983).
2. J. F. Allen, and P.J. Hayes, "A Common Sense Theory of Time", Proc. 1985 of the Ninth International Joint Conference of Artificial Intelligence, Los Angeles, CA.
3. U.S. Army TRADOC 1985 Draft Manual, "Army 21, A Concept For The Future, Appendix E, Intelligence and Electronic Warfare", Fort Haucha, AZ.
4. D. W. J. Chubb, "Advanced Spatial Problem Solving Methodology Using the Functional Binary Decomposition Spatial Representation", Proc. of the 1988 U.S. Army Science Conference, West Point, N.Y.
5. F. Hatfield, D. A. Madalon, and S. C. Williams, "An Artificial Intelligence-Based Paradigm for a Tactical Battlefield Situation Assessment System", The Analytic Sciences Corporation TR-5399-2, Arlington, VA., (1987).
6. R. V. Katter, C.A. Montgomery, and J.R. Thompson, "Imagery Intelligence (IMINT) Production Model", Research Report No. 1210, U.S. Army Research Institute for the Behavioral and Social Sciences, Alexandria, VA., (1979).
7. N. J. Nilsson, "Principles of Artificial Intelligence", Tioga Publishing Company, Palo Alto, CA., (1980).
8. R. Wilensky, "Planning and Understanding", Addison-Wesley, Reading, MA., (1983).
9. R. Wilensky, "KODIAK: A Knowledge Representation Language", Proc. 1984 Sixth National Conference of the Cognitive Science Society, Boulder, CO.

THIS PAGE LEFT BLANK INTENTIONALLY

Data Fusion in ESAU: Expert System Air Order of Battle Update

Prepared by:
Greg Gibbons, Jill Josslyn, Joyce Musselman, and John Russell
Systems Control Technology
2300 Geng Road
Palo Alto, CA

INTRODUCTION

ESAU (Expert System AOB Update) is an expert system command and control decision aid, currently under development for RADC/COAD. ESAU is intended to assist intelligence personnel in the Tactical Air Command Center (TACC) with the task of keeping track of the counts, locations and capabilities of enemy aircraft. This information is maintained in the Air Order of Battle (AOB).

An essential role in management of a modern air battle is keeping track of the enemy aircraft - where they are based, what weapons and support facilities are available to them, what sorties are being mounted from what airfields, and so on. This information, together with understanding of Soviet tactics and doctrine, enables the intelligence staff to recognize tactically significant developments in the air battle, and thereby to support the commander's ability to respond appropriately.

The air order of battle consists of a list of enemy military aircraft, including a count of aircraft by type, model, and country code for each airfield in the theater. The AOB is maintained by an AOB analyst, on the basis of a number of different kinds of intelligence reports. The AOB analyst's job is to maintain the AOB in as complete and correct a condition as possible, in a rapidly changing wartime environment. This is a data fusion process, requiring judgment and understanding, because the reports are often incomplete, uncertain, or even contradictory.

Keeping track of this information involves considerable amounts of what is called 'bean counting': counting, sorting, transcribing information from messages, etc. This bean counting is necessary to support the analysis, which leads to the valuable resulting information. Bean counting to some extent, and analysis especially, are largely data fusion - combining

information from a number of sources, extracting significance and meaning from large and diverse amounts of detail.

In current practice, these fusion processes are accomplished entirely manually. Many of the bean counting tasks are especially appropriate for automation; further, the tasks that involve some element of judgment, knowledge of enemy doctrine, etc., can be materially assisted by decision aids based on artificial intelligence technology.

ESAU will support the AOB analyst by assisting with many of the bean counting tasks, and by using expert system techniques to generate recommendations of updates to the AOB, based on events reported in incoming intelligence reports.

In this paper we describe ESAU and the AOB analysis process in terms of the C2 system of which it is a part, the kinds of intelligence inputs which it uses, the specific tasks performed by the analysts, and the role of ESAU in supporting the analyst.

BACKGROUND

ESAU is the next generation of the DAGR (Dynamic Order-of-Battle Aggregation Aid) program, also developed by RADC/COAD. DAGR was a proof of concept system that applied expert system technology to the problem of assisting an analyst to maintain an up to date Air Order of Battle estimate, based on incoming air track information and engagement reports.

In its formal evaluation, DAGR was found to enable untrained personnel to perform the AOB analysis role more effectively than trained analysts were able to without use of the aid. DAGR did much of the bean counting, and called attention of the user to appropriate changes to the AOB, so

that untrained personnel were able to do an effective job. DAGR also improved the effectiveness of trained personnel.

ESAU, currently under development, will be a free standing microprocessor-based aid. It will use real message traffic, and will provide a robust capability to assist in maintaining the AOB in realistic exercise and operational environments. ESAU will perform data fusion functions, including correlation of air track data and engagement reports, and assisting in recognition of tactically significant developments in the air battle.

The officers charged with maintaining the AOB are in the Intelligence section of the Tactical Air Command Center (TACC). Thus ESAU will provide assistance with an existing functional role in the context of the overall TACC. The structure and organization of the TACC is well established, as is the role of the AOB analyst within it. ESAU's development plan avoids the usual technologist's temptation to try to redesign the whole C2 system. Rather, ESAU addresses specific functions which can be improved with automation, leaving the overall flow essentially intact but more effective.

ESAU IN THE TACTICAL AIR COMMAND CENTER

To understand the role of ESAU, consider the overall structure and function of the TACC. The TACC is divided into two main sections: Current Operations and Current Plans. The intelligence section of Current Ops is ENSCE: Enemy Situation and Correlation Element; in Current Plans, the intelligence section is CID: Combat Intelligence Division.

Current Ops conducts today's war; this includes conducting air defense, responding to problems and opportunities as they arise in the air battle, and executing the planned air tasks. The plan is contained in the daily Air Tasking Order (ATO), and supplementary air tasking messages.

The ATO is a comprehensive order assigning aircraft and weapons from specific units to targets, with compositions of attack packages, time on target (TOT), tanker and EC support, etc. The ATO is produced on a daily basis in the Current Plans section of the TACC. In addition to producing the ATO, Current Plans is responsible for assessing enemy capabilities, and various orders of battle (OBs), including missile (MOB), electronic (EOB) and air (AOB). This capabilities assessment, together with an assessment of the enemy's probable course of action (PCA), is used to support the commander's staff situation briefing, to provide information on an informal basis to other TACC staff, and to support the planning of the next day's ATO.

The TACC operates on messages - some voice communication, some handwritten hardcopy, but much of it in electronic digital form. These

electronic messages are typically well formatted, and much of the important information is indeed contained in easily decoded fields. Significant information is also contained in free text fields such as AMPN (amplification).

Within a TACC, information is stored and presented mainly on wall maps, with some summaries and tabulations being hand generated in hardcopy. These wall maps are an essential element of the TACC, because they enable groups of people to discuss the situation in a way that small computer generated displays can not. There are wall maps for each major function: planned reconnaissance missions, enemy ground forces, enemy air defenses, etc. In particular, there are AOB maps, typically one showing airbases with aircraft assigned, the other showing plots of sorties and estimated enemy sortie corridors.

TACC operations at present are essentially manual. There are electronic message handling and management systems, and the TACC personnel themselves have produced some small application tools that run on the Air Force Z150s and Z248s. But current practice does not include any substantial automated support for understanding the meaning and importance of the messages, maintaining the OB databases, generating estimates and briefing materials, or any of the other myriad interpretation and bean counting tasks of the TACC staff.

The TACC is a large and complex organization, with many well-specified technical roles, of which AOB analysis is just one. For the remainder of this paper, we will concentrate only on the AOB role.

TASKS PERFORMED BY AOB ANALYSTS

ESAU is a decision aid for the AOB analyst. To understand how ESAU can help the analyst, consider the nature of the analyst's job. The role of the AOB analyst is to merge the incoming information in messages with his knowledge of Soviet tactics and methods, and with preexisting reference data, to develop an up to date understanding of the enemy's forces, capabilities, and intentions. To do this, he performs a number of specific tasks, as follows.

AOB analysts tabulate enemy sorties by day, by mission type, by base of origin, and possibly by other characteristics as desired. Enemy routes of flight are plotted on a wall map, in order to recognize sortie corridors.

The AOB analyst attempts to recognize tactically significant events, such as redeployments, or sequences of events which are to be expected based on Soviet doctrine. For example, deployment of the Mainstay (the Soviet equivalent

of our AWACCS) aircraft may be typically followed by more localized jamming, and then by a ground attack. Soviet doctrine is explicit and rigid, so that a prime objective is to anticipate their moves, deny their critical steps, and thereby reduce their effectiveness. Understanding of these sequences and of enemy corridors can lead to better air defense decisions as well as recognition of our vulnerabilities, enemy PCA, and targeting opportunities.

For example, in one exercise it was strongly desired to destroy or prevent the operation of a Mainstay aircraft. The Mainstay was defended by Foxbats, sufficiently so that our air to air missions were ineffective against it. One of the AOB officers observed that the Foxbats were based at a relatively undefended field, so that by attacking the Foxbats on the ground we could defeat the Mainstay. This plan conserved valuable air-to-air assets, and neutralized the Mainstay with a relatively economical bombing raid. This is an example of a targeting opportunity being recognized by an AOB officer.

Redeployments of aircraft from one base to another can be significant events: the aircraft themselves are the primary threats, and the primary targets. The base facilities only support

them. Deployment of new aircraft into the theater, or the positioning of new capabilities such as CBW capable systems, are important to recognize. Again, there are some predictable signs of impending redeployments - pre-positioning of fuel, weapons, and support facilities, for example, and recovery of aircraft at other than their base of origin. The association of killed aircraft with their base of origin is one of the main tasks of the AOB analyst, and provides the basis for accurate estimates of enemy dispositions.

In summary, the AOB's task is to perceive and understand enemy activity related to air strength.

This is a matter of judgment, training and experience, and requires the analyst to be able to think through the evidence and discuss it with other members of the TACC staff. In support of this judgmental responsibility there are a number of bean counting tasks which are currently executed manually, and which take up most of the AOB analyst's time.

SUMMARY OF THE AOB RELATED MESSAGES

Figure 1 shows the flow of AOB related information in the TACC. There are three primary

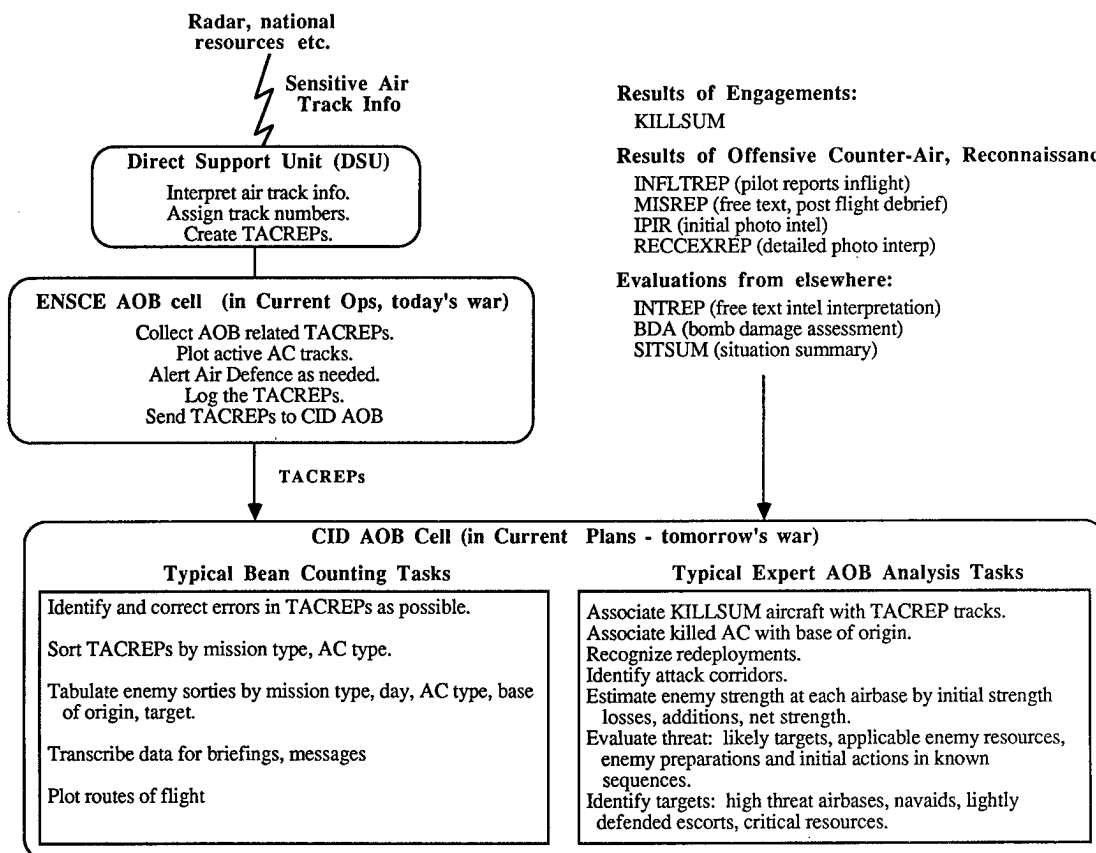


Figure 1: AOB Related Information Flow within the TACC

kinds of information about aircraft: air track information, results of engagements of enemy aircraft, and results of offensive counter-air (OCA) missions, which target aircraft and facilities on the ground.

Air track information is based mainly on radar and other tracking systems, and comes in from a specialized intelligence unit called the Distributed Support Unit (DSU). The DSU receives inputs from highly classified intelligence sources and national resources. The DSU processes the incoming information, manually sanitizes it to the SECRET level, and hand types it into messages called TACREPS (Tactical Reports). TACREPS contain air track information; their track numbering, and therefore the correlation of multiple observations with tracks, is highly reliable. TACREPS provide information such as time of observation, aircraft type and model, speed, heading, altitude, etc.

The second major kind of information about aircraft airborne is engagement reports. When either aircraft or ground defenses engage enemy aircraft, they report the results, including time, location, type and model, type of engagement, etc.

Engaged forces have no way of knowing about the information in the TACREPS; they merely report what they know from their engagement. Thus it falls to the AOB analyst to combine the information; in particular, one of the major responsibilities is to determine the base of origin for killed or landed aircraft. This is essential in order to know which base to decrement, and therefore what change has occurred in the enemy AOB. This information is also basic to recognition of redeployments and estimation of sortie generation capability.

Information about aircraft on the ground comes from flight crews of offensive counter-air (OCA) missions, from reconnaissance missions, and from national resources. This information relates to aircraft and facilities observed as damaged, destroyed, or otherwise changed.

These three kinds of information - track data, engagement results, and OCA results, are the main inputs for the data fusion process that is done by the AOB analyst. These three kinds of information are conveyed in the TACREPS, KILLSUMS, and reconnaissance reports, respectively. In the following sections we describe these messages and how they are used by AOB analysts and ESAU.

A DAY IN THE LIFE OF A TACREP

TACREPS are produced in the highly classified DSU. They contain air track information from sensitive sources, manually sanitized and distributed at the collateral level. They are sent

as messages to the Current Operations section of the TACC for immediate action. In particular, they are sent to the ENSCE Air TACREP desk, at teletype rate, 30 characters/sec. There the track's latest position, heading, speed and altitude are plotted and air defense is alerted as appropriate. Then the messages are logged. Collected messages are forwarded periodically to the Current Plans section of the TACC, to the CID AOB desk. These are hardcopy messages, forwarded via the 'tennis shoe net' - i.e., hand carried.

The AOB Officer in Current Plans receives the TACREPS, sorts them by mission type, plots their routes of flight on his wall map, and updates his tallies of enemy missions by mission type, base of origin, and aircraft type. He uses these tallies and plots to determine significant events and trends, to identify the airbases the enemy is using, and to recommend defensive measures and targeting opportunities.

Because TACREPS are manually produced, occasional spelling and other errors come through. TACREPS may be as much as 15 or 20 minutes behind real time. In a major air battle there may be as many as 3000 TACREPS per day.

OTHER MESSAGES

In addition to TACREPS, the AOB analyst makes use of a number of other types of messages. Some of these are:

KILLSUM (Kill Summary): This is a list of claims of destroyed aircraft. It is a well formatted message, each line of which is the word CLAIMS, followed by codes for the type of engagement, number of aircraft destroyed on the ground, number and location of aircraft engaged, and so on. It is the KILLSUM which provides the AOB officer with the results of engagements with enemy aircraft.

MISREP (Mission Report): This report is based on a quick aircrew debriefing. It provides early assessment of the mission's effectiveness, but may be less accurate than a later, more detailed assessment.

IPIR (Initial Photographic Intelligence Report): This is a quick interpretation of reconnaissance photography; again, it may be subject to correction based on later more detailed photo interpretation.

RECCEXREP (Reconnaissance Exploitation Report): This is the detailed result of a reconnaissance mission. RECCEXREPS are fairly well formatted, but have fields which are narrative in form, as well as the all-purpose AMPN (amplification) field, which is free text. For example: "no AOB noted"

THE ROLE OF DECISION AIDS FOR AOB ANALYSTS

ESAU will provide specific capabilities in support of the AOB analyst. In addition to those currently planned, there are opportunities for a number of advances which could materially contribute to the functioning of the TACC. In this section we discuss both the specific capabilities currently under development and the potential for the future.

ESAU Functionality

Figure 2 shows the functional flow within ESAU, illustrating the main functional capabilities of the system. ESAU will operate on TACREPS, KILLSUMS and RECCEXREPS. Each incoming message will be subjected to error detection and correction; where ESAU is unable to confidently deal with an erroneous message, user interaction will be solicited.

ESAU will associate each TACREP and each killed aircraft in a KILLSUM with the corresponding track in its internal blackboard database. The principal means of association for TACREPS is their internal track number, which is known to be

reliable. The association of killed aircraft with existing tracks will be accomplished by using expert rules for selecting the most appropriate candidate associations. These rules will compare aircraft types, geometric calculations based on observed course and speed in the TACREPS vs location and time of the engagement reported in the KILLSUM, and so on.

Expert rules will also be used in associating air tracks with base of origin. In some cases the TACREPS may contain this information; in other circumstances they may not, and the association of tracks with base of origin has to be made in the AOB cell. These rules will use similar considerations of reasonableness, involving the aircraft and support capabilities known to be at candidate bases of origin, course geometry compared to the location to the candidate base, etc.

Determination of the base of origin will enable ESAU to recommend changes to the AOB based on enemy losses in combat. Similarly, when aircraft land at other than the base of origin, this is evidence of a redeployment, since the Soviets seldom use recovery bases other than the base of origin except in emergency. ESAU will call the user's attention to the possibility of redeployments appropriately.

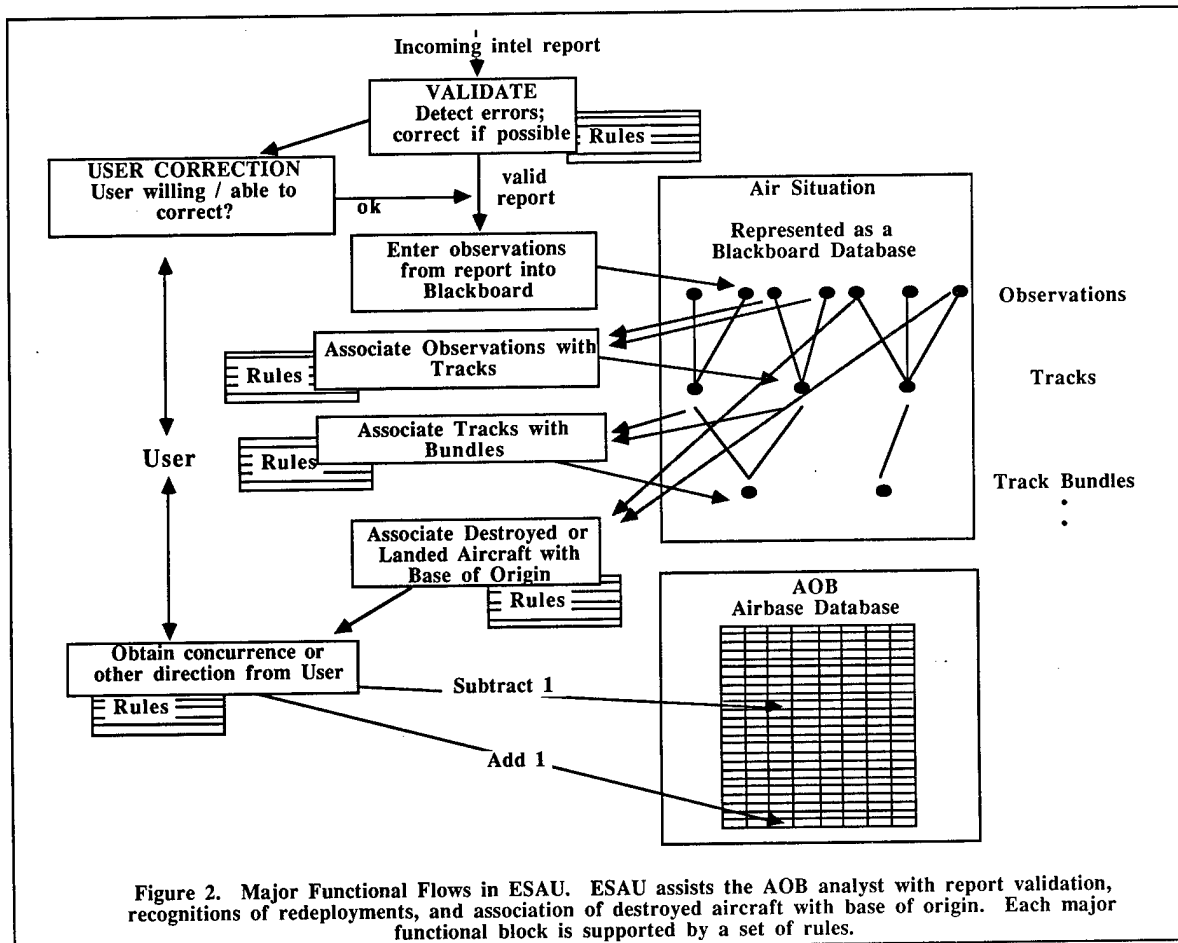


Figure 2. Major Functional Flows in ESAU. ESAU assists the AOB analyst with report validation, recognitions of redeployments, and association of destroyed aircraft with base of origin. Each major functional block is supported by a set of rules.

Expert rules for ESAU are not yet available, because ESAU is currently in the early design stage. However, the predecessor program, DAGR, is the baseline for ESAU. To provide the flavor of ESAU rules, we have included in Figure 3 some rules taken from DAGR. ESAU rules will be more comprehensive and less hardcoded, but their general character can be seen in the examples from DAGR.

RECCEXREPS will report, among other things, numbers of aircraft observed damaged or destroyed on the ground. Of course, such losses need to be reflected in changes to the AOB.

A major part of the AOB analyst's time is spent in manually sorting, collating, and tabulating AOB information. Much of the analyst's time is wasted in doing tedious, error prone clerical activity, rather than doing what he is trained to - analysis. For instance, at a recent exercise, over an hour was spent manually producing a list of enemy air assets sorted by six general role codes. The AOB included specific role codes for each aircraft, but the request was for them to be consolidated into six categories. This kind of 'bean counting' task constitutes a significant fraction of the analyst's workload. ESAU will provide database management support to allow much of this bean counting to be done automatically.

Future Possibilities for Automated AOB Decision Aid

The functional capabilities being built in ESAU are just the beginning of what automated decision aids could do for AOB analysts. Some of the major benefits would accrue merely because the process is computerized: once the messages are decoded, databases are established on line, and procedures and forms are implemented, standard data processing technology will provide substantial improvement in operational effectiveness of the AOB officer. Beyond that, the data fusion technology will enable the system to assist in tasks that require some degree of judgment, such as the recognition of tactically significant events and patterns of behavior.

Various bean counting summaries are currently produced manually, including tables of sortie rates by airbase and by mission type. TACREPS, which are printed on slips of paper, are physically sorted by mission type and aircraft type. Clearly these processes can be done electronically. Moreover, elementary graphics capabilities will be able to provide charts showing such things as sortie generation over time, which is not shown graphically at present.

Just the maintenance of the TACREP log would be a big improvement. AOB officers track enemy activities for PCA prediction; there is no

Example AOB Update Rules

EVENT	SITUATION	ACTION
Kill reported.	Base of origin (BOO) confidence greater than zero.	Subtract one aircraft of most likely type and make, etc. from the most likely BOO.
Landing reported.	Recovery base (RB) confidence greater than zero.	Subtract one aircraft of most likely type and make, etc. from the most likely BOO, and add one to the most likely recovery base.

Track/BOO (or RB) match.

REASON	CONFIDENCE
Distance from track to base > 75 km	No match
Distance from track to base < 75 km	(20-distance)/10
Aircraft type (possible) at base	+2
Aircraft type not (possible) at base	-2
Aircraft type unknown	0
Aircraft model (possible) at base	+2
Aircraft model not (possible) at base	-2
Aircraft model unknown	0
Aircraft country (possible) at base	+2
Aircraft country not (possible) at base	-2
Aircraft country unknown	0
(Track bearing - base bearing) < 30 degrees	+2

Kill/BOO Match

REASON	CONFIDENCE
Distance from track to base > 75 km	No match
Distance from track to base < 75 km	(20-distance)/10
Aircraft type at base	+2
Aircraft type not at base	-2
Aircraft type unknown	0
Aircraft model at base	+2
Aircraft model not at base	-2
Aircraft model unknown	0
Aircraft country at base	+2
Aircraft country not at base	-2
Aircraft country unknown	0
Means of destruction=Fighter	+2
Attacker/target angle < 30 degrees	+1

Kill/Track match

REASON	CONFIDENCE
Distance from track to base > 75 km	No match
Distance from track to base < 75 km	(20-distance)/10
Aircraft type at base	+2
Aircraft type not at base	-2
Aircraft type unknown	0
Aircraft model at base	+2
Aircraft model not at base	-2
Aircraft model unknown	0
Aircraft country at base	+2
Aircraft country not at base	-2
Aircraft country unknown	0
Track reports count drop	+2

Figure 3. Example Rules from DAGR.

formal logging process. AOB officers keep written notes, and use them in briefings. The availability of machine produced logs and graphics would not only relieve the analyst of much tedious manual work, but would eliminate a potentially serious source of errors, since the final product is now the result of a number of manual transcriptions.

Using a TACREP history, it will be possible to produce graphic outputs of track corridors, thereby assisting in the recognition of patterns of enemy operations.

The AOB analyst must consider some normal expectable sequences of events rather than consider the arrival of each individual TACREP independently. For example, an offensive counter-air attack will be quickly followed by an INFLTREP (inflight report), which is based on the aircrew's verbal report of results while still airborne. When the aircrew is debriefed, on landing, the resulting message is the MISREP, which is less timely but more comprehensive and accurate than the INFLTREP. Later, if reconnaissance is ordered, there will be a similar sequence consisting of the IPIR and RECCEXREP. The result of all this is the battle damage assessment (BDA). Now all of this takes time, and during the interval the AOB analyst knows that the AOB is out of date, and may know something about the change to be expected. One hears remarks like "we haven't gotten our BDA yet, so we know there are fewer airplanes but we don't know how many fewer." Thus there is a role for the interpretation of the current estimates, based on knowledge of predictable sequences of events.

An automated aid should help prepare messages. For example, "Enemy Losses and Replacements" is currently produced manually; it is simply a tabulation showing type of aircraft, initial number, number destroyed, number of replacements, resulting number available. Other messages contain free text summaries, followed by tabulations; incorporation of automatic tabulations via an integrated message preparation aid would materially help the users.

CONCLUSIONS

Fusion is an integral part of the AOB function in the TACC. The CID Intelligence Officer's role is just fusion: the combination of diverse forms of incoming messages and other information to form an up to date, detailed, well supported estimate of enemy dispositions, capabilities and intentions.

Fusion is accomplished mainly by manual methods. At present there are essentially no electronic aids available to the AOB officer, other than message handling systems. Moreover, since the wall map and grease pen are central to the officers' ability to discuss the situation effectively with each other, it is likely that at

least the wall map will remain a manual function for some time.

Fusion is substantially supported by easily automated bean counting operations. Database functions of sorting, counting, updating, correlating, and charting can easily replace significant amounts of manual labor, leaving more think time for the officers, reducing the required staff size, and eliminating manual errors of omission and transcription.

Expert system technology can support the judgmental aspects of fusion in the TACC. Introduction of new weapons into the theater, redeployments, navigational corridors, changes in patterns of behavior, all can be called to the attention of the user. Using expert system techniques, the system can suggest possibilities and weigh alternatives in much the same way that the user himself does, and thereby function as a supportive colleague.

References

Adelman, L. and Crowley, J., "Test Analysis Report for the Dynamic Air Order-of-Battle Aggregation Aid (DAGR)," PAR Technology Corporation Contract Report #83-96, August 1983.

Gibbons, G., "Expert System Air Order of Battle Update (ESAU)," Systems Control Technology, Inc., Proposal, March 1987.

Gibbons, G. and Riley, K., "An Expert System for Air Order of Battle Maintenance in LOCE," Systems Control Technology, Inc., White Paper, May 1986.

Markosian, L.Z. and Riemenschneider, R.A., "Dynamic Air Order-of-Battle Aggregation Aid: Systems Specifications," PAR Technology Corporation Contract Report #83-133, October 1983.

Markosian, L.Z., Riemenschneider, R.A. and Rockmore, A.J., "Dynamic Air Order-of-Battle Aggregation Aid: User's Manual," PAR Technology Corporation Contract Report #83-102, September 1983.

Riemenschneider, R.A., Markosian, L.Z. and Rockmore, A.J., "An Expert Consultant for Dynamic Force Assessment," The Conference Record of the ACCSC, October 1983.

THIS PAGE LEFT BLANK INTENTIONALLY

FROM LABORATORY TO THE FIELD: PRACTICAL CONSIDERATIONS IN LARGE SCALE DATA FUSION SYSTEM DEVELOPMENT

Gregory Fox

TRW Federal Systems Group
2751 Prosperity Ave., FVA5/2179
Fairfax, Va. 22031
(703)-876-8014

Stephen Arkin

Space and Naval Warfare Systems Command
Department of the Navy, Code PMW 161-1
Washington, D.C. 20363-5100
(202)-692-7990

During 1988, the first phase of the OSIS (Ocean Surveillance Information System) Baseline Upgrade (OBU) system is being deployed to the five OSIS data fusion and correlation sites. As a major upgrade of a currently operational capability, a conservative set of laboratory and field-proven data fusion correlation and tracking algorithms has been implemented in this system.

Data fusion systems can be categorized based on four attributes:

1. Variety of sensor input - "similar sources" (e.g. receipt of data for correlation from many acoustic sensors) or "dissimilar sources"
2. Closeness to the sensor(s) - "embedded" (existing as a part of, or in close proximity to, sensors; e.g. aircraft cockpit correlation of radar and EO sensors) or "autonomous" (away from the sensor)
3. Destination of fused results - "localized receptor" (results do not go beyond immediate system/operators) or "receptor/transmitter" (results are transmitted beyond the immediate fusion system).
4. Sensor sample rate - "oversampled" (updates at twice target maneuver rate) or "undersampled" (updates less than maneuver rate)

The OBU system as a "dissimilar source", "autonomous", "receptor/transmitter", "undersampled", correlation system must operate in the most complex data fusion environment. This paper will explore some of the issues encountered integrating disparate algorithms operating on dissimilar (both in data content and security classification) sensor data under the roof of a single system. In addition to algorithm implementation, complexities in the areas of message handling, analyst interaction, data management, system recovery, testing, handling of multi-level security, and throughput performance had to be overcome in order to turn data fusion theory and proven algorithms into an integrated operational system.

The paper will also discuss the flexibility required in the operation of the fusion algorithms as a result of both the dissimilar nature of the sensors and the differing nature of the fusion problem at geographically separate site locations. The OBU system architecture not only provides growth in the hardware and software dimensions, but also supports flexibility and growth in the algorithm dimension. The currently implemented system provides both serial (execution of a sequence of algorithms on a single incoming contact report) and parallel (execution of different sequences of algorithms for different incoming contact reports) adjustment of the fusion algorithms using a process called "channelization."

FROM LABORATORY TO THE FIELD: PRACTICAL CONSIDERATIONS IN LARGE SCALE DATA FUSION SYSTEM DEVELOPMENT

Gregory Fox

TRW Federal Systems Group
2751 Prosperity Ave., FVA5/2179
Fairfax, Va. 22031
(703)-876-8014

Stephen Arklin

Space and Naval Warfare Systems Command
Department of the Navy, Code PMW 161-1
Washington, D.C. 20363-5100
(202)-692-7990

OVERVIEW

This paper describes theoretical and practical aspects of conceiving, building and deploying a large data fusion system. The discussion is presented in the context of development of the Navy's Ocean Surveillance Information System (OSIS) Baseline Upgrade (OBUS).

DATA FUSION SYSTEM EVOLUTION

The generic evolution of a data fusion system development is shown in Figure 1. The flow within the figure and the influence of practical considerations on the evolution from algorithms and architecture to an operational system are the basis for this paper. This evolution starts with consideration of the fusion environment (called "Nature of Correlation Problem" in the figure) and with "Mission and Operational Constraints". Three components of the data fusion approach are developed by applying "Technology" to the challenge of the environment and the operational mission: algorithms, data fusion architecture, and system design. The "Nature of the Correlation Problem", "Mission and Operational Constraints" and "Technology" subjects in Figure 1 represent knowledge about the fusion problem and potential methods of solution. The three boxed items derived from them represent the initial technical definition of the solution.

Once algorithms are selected, architecture is determined and system design is complete, the system must be built and tested. A number of practical "Development

Realities" either are discovered, or, if previously known, begin to have more influence. As represented in Figure 1, these realities and the resulting "Development Consequences" affect the realization of the final system by transforming or modifying portions of the design to achieve a workable "Operational System".

The major section headings of this paper will be keyed to Figure 1. With the exception of "System Design" and "Operational System", which are topics too large for a paper of this length, each subject shown in the figure will be discussed. At the end of this paper, some planned and possible future directions for the OBUS system will be discussed.

NATURE OF THE CORRELATION PROBLEM

Four environmental factors can be used to categorize a data fusion system: source, location, destination and information. Together they define the "nature of the correlation problem." Figure 2 addresses three of these factors. In the figure, fusion system S1 receives information from a single type of source (or sensor) while system S2 receives information from more than one type of source; this is the source factor. S1 is a "similar source" system, S2 a "dissimilar source" system. The complexity in several areas of the fusion system can be directly related to the source factor: preprocessing before data fusion, data fusion algorithms, and analyst tools. In addition, if the system must pass fusion results to external users, dissimilar security classifications between sources can add to the difficulty of producing reports.

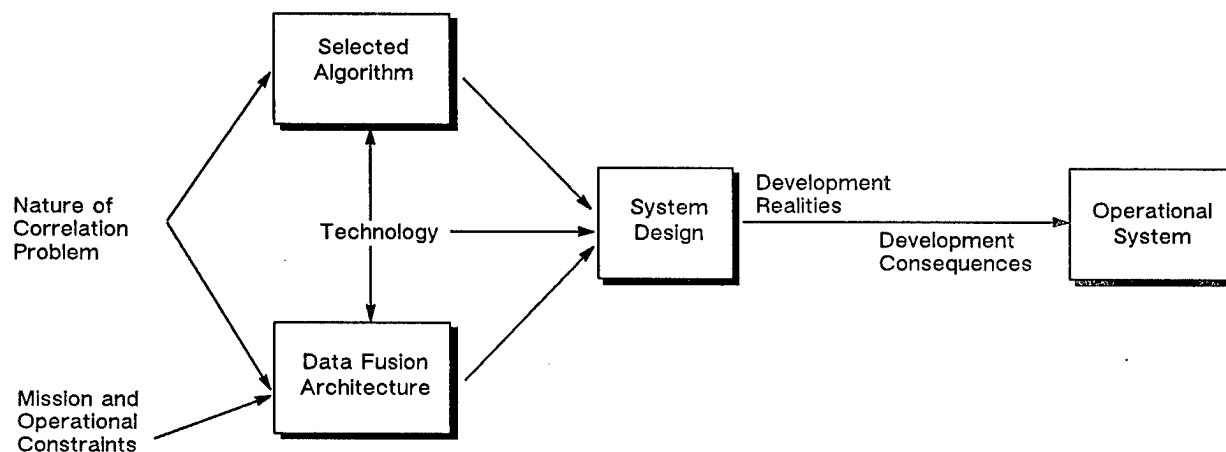


Figure 1. Evolution of Data Fusion System Development

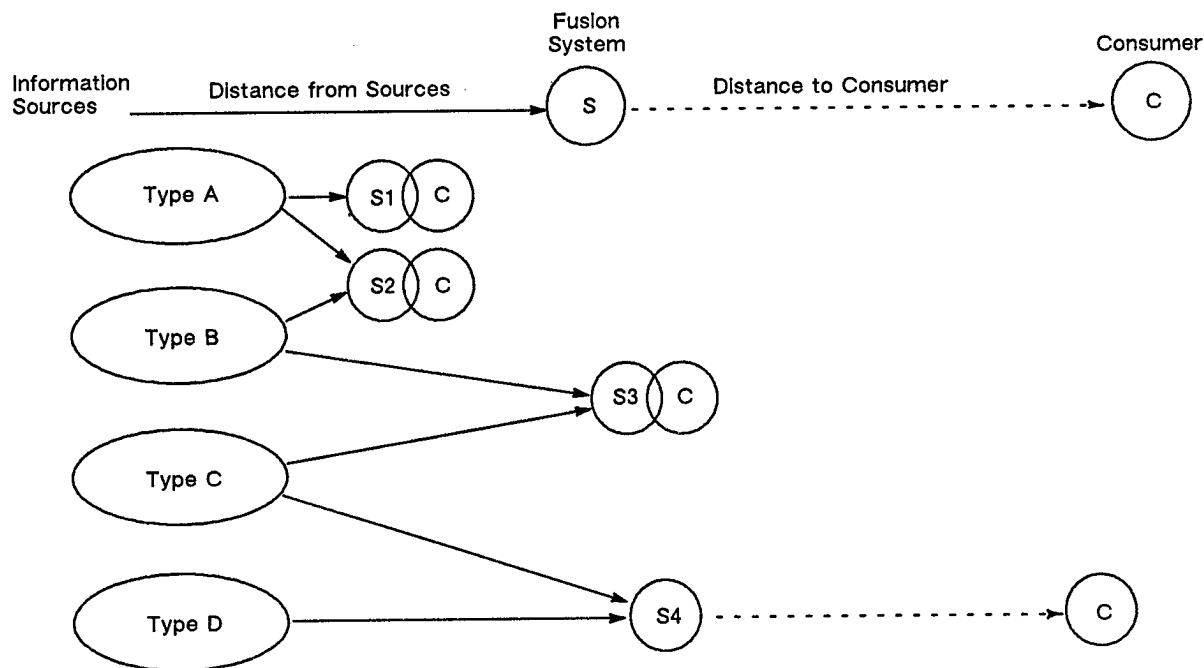


Figure 2. Relationships of Data Fusion Systems with Data Sources and Consumers

The second factor, the location factor, addresses the proximity of the fusion system to the source. Proximity can have a physical dimension, a functional dimension, or both. In the figure, both S1 and S2 are collocated with the sensors and represent systems "embedded" with the sensors. These systems deal with sensor information in a raw or partially processed form, and receive timely and time-ordered data input. In contrast, S3 is an "autonomous" fusion system located away from the sensors. It receives processed and translated (typically into standard communication messages) data derived from sensor information; the data is usually not as timely as that received by the embedded system. The amount of preprocessing required before data fusion takes place and the form of the data fusion algorithms (information may arrive out of order) are related to the location factor.

For systems S1, S2 and S3 the consumer of the fusion results resides at the fusion system, while for S4 the consumer is at another location. This third factor, the destination factor, can be used to categorize a fusion system as either a "localized receptor" or a "receptor/transmitter". The destination factor determines the need for capabilities in the fusion system to post-process fusion results into reports that can be sent to consumers. It can also impact system data management by requiring special security handling in fusion systems having dissimilar security classification source input.

The final factor, the information factor, deals with the quality of input information. Both accuracy of information and sample rate with respect to target maneuver rates must be considered when selecting data fusion algorithms. Adequately or oversampled systems can use classical correlation and tracking algorithms. "Under-sampled" systems must rely on more involved means in an attempt to fill gaps in the information. This results in the need for sophisticated fusion algorithms and analyst

tools for resolving inconsistencies. The information factor is an important driver in developing the algorithms; the source, location and destination factors can be as important when developing a system.

The OSIS (Ocean Surveillance Information System) Baseline Upgrade (OBU)

During 1988, the first phase of the OBU system is being deployed to the five OSIS data fusion and correlation sites. As a "dissimilar source", "autonomous", "receptor/transmitter", "undersampled" correlation system, it must operate in one of the most complex data fusion environments. An additional factor influencing the OBU development was an existing operational capability: the OSIS Baseline System (OBS). It necessitated an approach that did not cause a major perturbation in the ability of the sites to continue operations. To avoid disruption of site operations from algorithm induced anomalies or failures, a conservative set of laboratory and field-proven algorithms based on existing technology was selected for the OBU system. The challenge of the data fusion environment was addressed in the OBU architecture; several algorithms were combined in a flexible and synergistic fashion.

In the case of the OBU system, the "dissimilar source", "autonomous", "receptor/transmitter" environmental factors are established from the OSIS Mission and Operational Constraints (discussed in the next section). The final factor, the quality of the sensor data, is common to most autonomous ocean surveillance systems. The information is generally greatly under-sampled, of varying degrees of accuracy, often received in a different order than acquired, and representing knowledge about disparate attributes of the target platform. This disparity can be illustrated by considering the information that might be received for a submarine. One

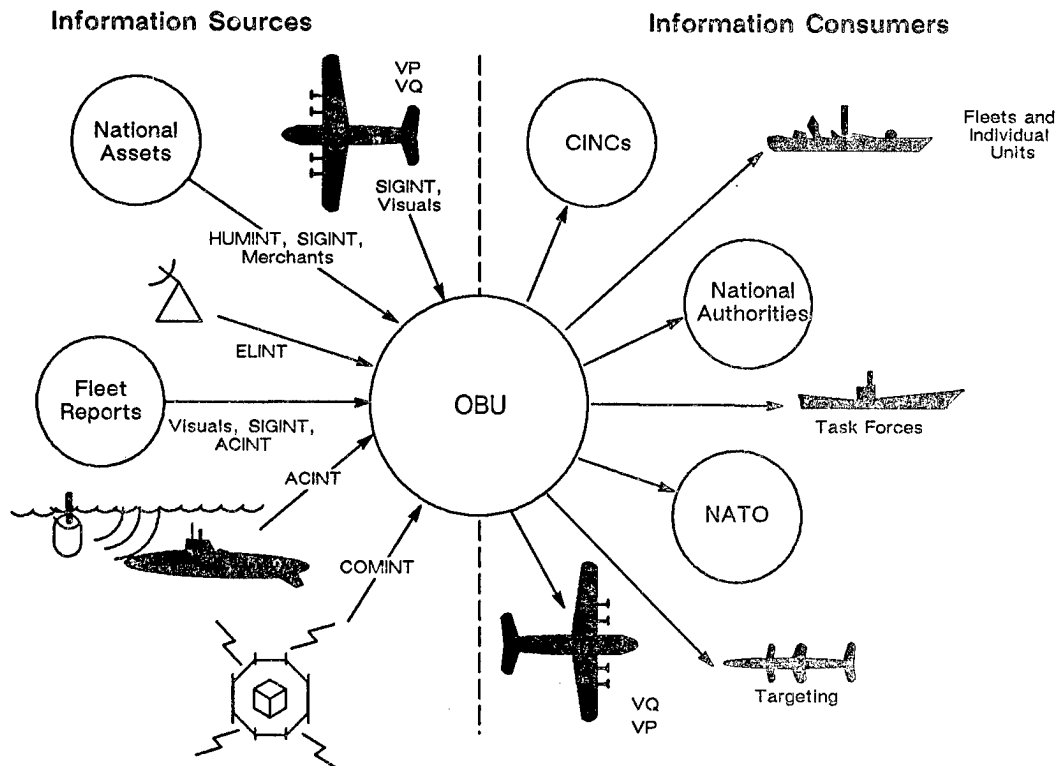


Figure 3. OBU Services a Variety of Sources and Consumers

report (called a contact) might contain a location and platform identification (visual sighting). It might be followed by a contact containing acoustically derived parameters and a line of bearing (ACINT), followed by a location estimate and measurement of radar parameters (ELINT), followed by a location estimate and a radio call sign (SIGINT). This information must be combined to provide an overall picture of the characteristics, location time line and identity of the specific platform.

OSIS MISSIONS AND OPERATIONAL CONSTRAINTS

The Ocean Surveillance Information System (OSIS) is an essential element of the Navy Command and Control System (NCCS) ashore. The Navy Command and Control Plan states the OSIS mission as "receive, process, and disseminate timely all source ocean surveillance information on mobile targets of interest, above, on, and under the oceans, to the Navy and other services at all levels of command." There has also been increasing interest in the support of Navy power projection missions through monitoring coastal zone land-based targets. There is a large number of information sources for the targets of interest and, within the mission definition, a large number of potential consumers of the results of fusion. As a result, the OBU system must receive and process positional and electronic intelligence reports for surface ships, submarines, aircraft, and points ashore. It must support the correlation, assessment, and evaluation of the reported information, and generation of the necessary Integrated Ocean Surveillance Products (OSP) to support both afloat and shore in-theater levels of command. The OSP includes both scheduled and event-by-event (EBE) reports. Figure 3 illustrates the variety of sources and consumers the OBU system must service.

The OSIS is a distributed system which is geographically oriented to support fleet commanders and afloat forces. As a result, the OBU system is being deployed to five operational sites, each having somewhat different availability of sensor data and emphasis within the overall OSIS mission. As an example, one site might place first emphasis on air platforms while another might place greater emphasis on afloat platforms. This emphasis is subject to change depending on the tactical situation. In supporting multiple sites, each performing multiple functions, the OBU system must provide the flexibility to support multiple analysts doing similar but different tasks.

Several key program level objectives derived from the OSIS mission guided the OBU development and were significant factors in defining the data fusion approach: provide a multi-source integrated locational data base, provide automated event-by-event reporting with no human in the loop, and allow the integration of data of multiple security levels and classifications, including Sensitive Compartmented Information (SCI). An additional programmatic objective was to establish a system backbone to support future growth. That growth may occur in several dimensions, from increased quantities of input data and increased variety of input sources, to more specific handling of the characteristics of important subsets of the target population. By developing a robust and flexible OBU system, relatively modest additional investment can evolve the system to meet changing environments and emphasis.

Many of the mission and program objectives can be summarized in a single primary concept: a single data base consisting of fused information on all platforms, from all sources, at all security levels that is available both for display and analysis by OSIS analysts, and for transmission in a tailored form to consumers of the OSP. It is this concept which guided the definition of the OBU system. Within a given operational site, there also exist

several small systems currently accomplishing portions of the overall ocean surveillance job. These systems exist as recently developed prototypes which have unique capabilities or applications. Combination of these capabilities with the OBU system to produce an integrated picture for the analyst is a part of this primary concept.

TECHNOLOGY

Referring to Figure 1, once the Nature of the Correlation Problem and the Mission and Operational Constraints are defined, technology is applied to develop algorithms, architecture and a system design. The technology base for the fusion system includes hardware, algorithms, and software engineering disciplines. In the case of OBU, no hardware or algorithm technology development was undertaken. Emphasis was placed on the exploitation of existing technologies within the framework of a unified system. New software engineering approaches were required for the development of a multi-level security accreditable large-scale software system. Existing technology for control and coordination of application processing on a set of local distributed computing nodes was applied to solve the combined need for very high system availability and support of multiple analysts in a multi-task environment.

SELECTED ALGORITHMS

The goal of algorithm selection in a data fusion system is to use the best algorithms. The factors that determine "best" vary from system to system. Given the disparity of input sources and platform types the OBU system must handle, there is no single algorithm that can correlate all of the information; algorithm selection was the determination of a synergistic set of algorithms. Information received by the OBU system is in the form of contact reports containing one or more of the following categories of information:

1. Unique Attribute - data which specifically identifies either a platform (like ship name) or a track (this report is one of a series belonging to the same platform)
2. Physical Data - items like ship class or country which provide partial identification of a platform
3. Parametric Data - measured parameters relating to a platform; in the OBU system these refer to measured radar parameters (ELINT)
4. Location Data - estimates of location with an associated containment ellipse. These data items occur in reports of the previous three categories, but can also be received alone.

These four categories of information are used within the OBU system in a hierarchical fashion, processed in the order listed. The only exception to the ordering is early screening of platform tracks by location; this avoids attempted correlation of a contact report to platforms not capable of being at the contact location at the reported time. The result of ordered processing is sequential application of separate correlation algorithms for unique attribute, parametric, and spatial information, with the use of physical and location information to eliminate incompatible correlation candidates.

Unique Attribute Data

Rules are used for matching this information to determine contact correlation to platform tracks, or for

elimination of platform tracks from further consideration. The rules were developed by examining other systems and studying message traffic; they allow the OBU system to take advantage of processing performed at the sensor (which can be as simple as reading a hull number off a ship through a pair of binoculars or as complex as tracking a target through processing radar returns). Although the individual rules are straightforward and do not, in the normal use of the terminology, constitute "algorithms", layering of this correlation process on top of the other correlation algorithms is an important aspect of the overall approach. This apparently simplistic correlation method is closely coupled to the reporting rules and conventions enforced in the reporting system. Changes in message formats, management of platform identifiers, or the existence of exceptions to defined meanings of message fields all serve to perturb the correlation rules which use unique attribute information.

The initial list of unique attributes for correlation or track candidate elimination used in the OBU system is given in Table 1. As a consequence of reporting conventions and sensor management of these attributes, not all can be treated uniformly in the correlation process. This distinction is indicated in the table under the columns labeled Number of Values and Qualifiers. Unique attributes having one value and no qualifiers are sufficient to cause correlation of a contact to a track in the case of a match, or elimination of the track from further consideration (for parametric and spatial correlation) in case of a mismatch. Attributes having more than one value and no qualifiers can cause correlation on a match, but do not disqualify the track from further consideration when there is a mismatch. Attributes having "plus" qualifiers are treated as concatenations of the attribute and the qualifier; this concatenation produces a combined value for both correlation and track elimination purposes. For example, a contact report with Pendant Number 805 and flag US will correlate with any track having the same Pendant Number/flag combination. All tracks having either a different Pendant Number or flag are disqualified from further consideration. Attributes having "with" qualifiers are treated as concatenations for correlation purposes, but track disqualification occurs only when two different values of the attribute occur for a given value of the qualifier. For example, a contact report with Raid Number 123 and command XYZ will correlate only with a track having the same Raid Number/command combination. Tracks with differing Raid Numbers will be disqualified for further consideration only if they have a command value of XYZ.

Table 1. Initial OBU Unique Attribute Parameters

Attribute	Number Of	
	Values	Qualifiers
NOSIC ID	1	None
(Radio) Call Sign	5	None
Ship Name	1	None
Pendant Number	1	Plus Flag, With Type
Precor Track Number	3	With Command
Raid Number	3	With Command
Trademark	3	With Command
PIF Code	3	With Command

The initial unique attribute definitions for correlation and their handling will be modified as dictated by experi-

ence, new data sources or changes in reporting conventions.

Physical Data and Consistency Tests

Physical data is used to eliminate incompatible correlation candidates. All candidate platform tracks having physical attributes (flag or country code, platform category, platform class, platform type) inconsistent with the contact report under consideration are eliminated from consideration for parametric and spatial correlation. All successful unique attribute correlations are also tested for physical data consistency; conflicts are reported to an analyst for resolution. In addition to the physical data, geofeasibility (determined from location and maximum platform category speed) and allowable platform emitter complement (determined at the ship class level from a stored table) are used in similar consistency and elimination tests.

Parametric Data

ELINT measurements of radar parameters represent the majority of the parametric information received by OBU. The current OBU system uses only this type of parametric data; acoustical and other types of parametric data are not processed for correlation purposes. Almost all ELINT parametric data received at an OSIS site has undergone preprocessing by the source to determine the type of radar reported; the radar type is identified by the SEDSCAF or ELINT Notation (ELNOT). Algorithms implemented in OBU, therefore, depend on the (correct) existence of the ELNOT for processing the parametric data.

The ELINT processing algorithm selected for OBU was developed from research performed at NOSC during the early 1980's. This algorithm, TERESA (Target Evaluation and Recognition by Extraction of Statistical Attributes), is a "second generation" ELINT correlation algorithm that uses a statistical distribution rather than range-oriented parametric tracking process.

First generation ELINT correlation algorithms are based on parametric and spatial trend tracking, using analyst managed parametric ranges and system derived statistical means to screen and score candidates. These algorithms evolved through a series of prototype efforts (NEAT - Naval ELINT Analysis Tool, BELT - Basic ELINT Tracker, POST - Prototype Ocean Surveillance Terminal) which used experiences with the initial algorithms and overlaid field developed heuristic rules on the analytical aspects of the correlation process.

TERESA uses a completely statistical approach to ELINT correlation. This algorithm assumes that the parameters of an emitter observed through independent measurements follow a probability distribution; correlation decisions are mathematically derived consistent with the assumed statistical model. TERESA initially uses default parametric distributions derived from analysis of observations of specific categories and versions of radars. The algorithm allows the parameters that characterize the distributions to evolve as samples are correlated to specific emitters. In this manner, the system "learns" the characteristics of specific emitters, eliminating the detailed management of ranges for scoring and correlation required by first generation algorithms.

Each generation of ELINT algorithms has its supporters and evidence seems to indicate that each out

performs the other in certain circumstances. References 1 and 2 contain a discussion of these algorithms and offer some comparison data. From these algorithms has evolved DUET (Developmental Unified ELINT Tracker) which incorporates both NEAT and TERESA algorithms (Reference 3). Data is routed to one or the other of the algorithms based upon its characteristics. It has not been shown that DUET would offer improved performance over TERESA in the OBU environment; this could be because superiority of the NEAT or TERESA algorithm may depend on the specific correlation situation rather than the characteristics of individual incoming data items. More experience with, and evaluation of, TERESA and DUET in operational environments is necessary to make this determination.

Location Data

Processing location data is one of the most mature forms of correlation processing because it is applicable to a wide variety of applications. Spatial consistency between a reported location and the estimated or projected position of a candidate platform is basic to many correlation systems. Determination of estimated or projected positions through the use of tracking techniques is a feature of systems that use location data to correlate moving targets. Many contact reports received by the OBU system do not contain unique attribute or unambiguous parametric information and therefore must be correlated based primarily on location. Possible track projection techniques vary from dead reckoning the last reported position to Kalman filter based projection. Because reports at an "autonomous" correlation system do not necessarily arrive in observed order, the selected method must accommodate projection to times before or after the most recent update to the candidate platform position.

A sophisticated spatial processing algorithm known as the Maneuvering Target Statistical Tracker (MTST) has been developed for tracking Naval ships (Reference 4) and proven through a series of demonstrations. MTST, as implemented in the OBU system, uses a four state (latitude, longitude, speed, and heading) model of the platform's estimated location including uncertainties associated with the estimate. This estimate is derived from a combination of position only and position/velocity contacts. The model assumes that uncertainties for both reported position and velocity are known or can be estimated. The algorithm also incorporates a motion model for tracking platforms that can maneuver between observations. The Integrated Ornstein Uhlenbeck (IOU) motion model is applicable to "undersampled" correlation systems. In the model, platforms are assumed to have a characteristic speed and a typical time between maneuvers; projected velocity is adjusted to account for possible maneuvers.

The MTST algorithm contains an additional feature that supports synergistic combination with Unique Attribute and ELINT correlation algorithms: post correlation aberrant point detection. In the OBU system approach to correlation, contact reports are processed against existing platform tracks yielding one of three correlation results: hard correlation to an existing track, failure to correlate to any existing track, or ambiguous correlation to more than one track requiring resolution by an analyst. This is known as the "single hypothesis" approach; a correlation decision in the OBU system can only be changed by manual intervention. This allows the possibility that a report having a poor spatial fit to a track will correlate because of unique attribute or ELINT data. A poor spatial fit could modify the track so that the projections for sub-

sequent correlation opportunities result in failure to properly correlate. A spatial miscorrelation can have the same effect. The MTST algorithm corrects this situation by post correlation aberrant point detection. This "soft" correlation feature allows the location component of a contact report to be ignored in the update of estimated position while incorporating the other information into the track. The location component can be determined aberrant and removed either for a newly correlated contact or for a previously correlated contact that is not consistent with subsequent correlations.

A current debate in the correlation community concerns the effectiveness of single hypothesis versus multiple hypothesis approaches to correlation. Multiple hypothesis approaches can defer making a "hard" correlation decision until the receipt of new and more definitive data, and can retroactively undo a prior "soft" correlation of a report to a track by considering new data. The potential advantage of such algorithms is a reduction in the number of ambiguous correlations requiring analyst resolution. The disadvantage is the immaturity of the approach in several practical aspects. The initial performance barrier to this approach has been addressed in recent theoretical and practical studies: how to efficiently "prune" the list of alternate scenarios retained to keep it within the bounds of current processing capability (Reference 5).

Even assuming that recent work has solved the pruning problem, there remain technical concerns inhibiting use of multiple hypothesis algorithms in an OBU-like environment. These concerns relate to "receptor/transmitter" correlation systems that report results to distant consumers. In this environment, computer resources are expended solving the security situation that results from combined fusion of multiple level security information and automatic reporting of results to external consumers. Even optimally pruned multiple hypothesis approaches require significantly more computer resources than do single hypothesis approaches. In addition the "pruning" aspects of multiple hypothesis systems make their consumption of computer resources, and therefore their timeliness performance, statistical rather than deterministic. Quantifying throughput performance before the system is deployed to the field becomes an extremely difficult task.

Additional concerns about multiple hypothesis algorithms deal with "changed calls" and recovery from failure. "Receptor/transmitter" correlation systems must report correlation decisions externally, often in real time. In the OBU system important ambiguities are resolved using the analyst's best judgement. This judgement can include information not available to the computer. Resolution of less important ambiguities can be deferred until more important ones are resolved. A multiple hypothesis system theoretically treats all ambiguities equally, awaiting further reports before making "hard call" decisions. Its reporting strategy must decide either to only report "hard call" correlations or to also allow reporting of most probable ambiguous correlations. The first strategy requires close analyst review of ambiguous correlations to decide when to force external reporting. This review can negate much of the potential advantage of these algorithms in off-loading work from the analyst. The latter strategy requires philosophies and mechanisms (on the part of both the fusion system and the product consumer) for an increased quantity of changed correlation messages to the consumer.

Recovery from failure (power, hardware or software) is always difficult in systems which require accurate retention of data and must meet throughput performance requirements. The problem of complete recovery of both the data base and the current processing report stream appears to be much more complex with a multiple hypothesis approach. Database recovery after failure was a major challenge in the single hypothesis OBU development. The impact of the journaling of updates to ensure recovery required iterative design efforts to meet throughput requirements. Journaling is necessary due to the "undersampled" nature of the sensor data reaching the OBU system; it cannot afford to lose an input report because another report on that platform may not be received for some time. "Brute force" approaches to recovery, like restarting the processing of reports from an input log, have to be avoided in a "receptor/transmitter" system so that communication circuits and consumers are not flooded with duplicate messages. Further discussion of multiple hypothesis approach issues can be found in Reference 6.

OBU DATA FUSION ARCHITECTURE

As shown in Figure 1, the OBU data fusion architecture was derived from Mission and Operational Constraints, the Nature of the Correlation Problem and Technology. This architecture is the result of five principles which guided its definition.

Principles of the Architecture

The first principle is that the architecture must allow the flexibility to tailor the correlation process to a site-specific environment. The five OSIS sites may have different types of platform targets, mixes of sensor data (a sensor which is a prolific provider of information to one site may be a meager provider to another), consumer reporting needs (some consumers need timely event-by-event reports while others need only periodic summaries), and priorities within the broad overall mission guidance.

The second principle is that the architecture must provide a backbone for future upgrades; it must be extensible. Extensibility in quantity of data processed and number of analysts supported is provided by the distributed hardware and network control software. The OBU data fusion architecture is also extensible in the algorithm dimension. The current algorithms can be adjusted to process different subsets of the data in different fashions without software change. The architecture also supports the insertion of new algorithms that are optimized for processing subsets of the data without significantly perturbing the existing algorithms.

The third principle is the adoption of a "divide and conquer" philosophy to achieve throughput performance. Envisioned as an all-source, all-target data fusion system, OBU was designed with the knowledge that its track database could become large. The combination of ship, air and land targets could result in many thousand targets of interest. Additionally, the use of all-source information in support of correlation raises the possibility of significant fragmentation of platform tracks. Data from sensors reporting poorly identified or ambiguous information can start tracks that cannot be extended or confidently correlated to nearby tracks. As a result, the system must protect itself from the performance implications of a large number of tracks. Two forms of protection are provided: a site adjustable method called "channelization" which avoids comparing all received

contacts to all tracks in the database, and a flexible scheme for establishing initiation strategies that can control the number of new tracks started.

The fourth principle is that the system must protect the correlation process from "bad" report data. Sensors and operators reporting observations make mistakes; unless detected these mistakes can corrupt the correlation data base. The OBU system implements aggressive consistency checking, notifying the analyst to resolve inconsistencies detected during correlation. The philosophy is that it is better to involve an analyst to resolve an ambiguity caused by bad data than to allow the bad data to diminish the possibility of future successful correlation.

The fifth principle results from system availability requirements and the mismatch between analyst and automatic software decision rates. A distributed hardware and software architecture not only eliminates single points of hardware failure but also decouples the analyst's decision process from that of the automatic software. The analyst is given time to reach correlation decisions without significantly slowing down the automatic software. The automatic software is able to make decisions at its natural rate, and to provide the analyst information necessary to keep him current.

Definition of the Architecture

The OBU correlation architecture is divided into two parts: automatic and interactive. The automatic portion performs correlation and tracking based on the algorithms previously identified. Results of successful correlation and ambiguous situations are communicated to analysts, operating at independent work stations, ac-

cording to both predefined and ad hoc criteria. Predefined updates are by automatic update to the work station of tracks previously retrieved by the analyst and "work lists" containing ambiguity notifications while ad hoc updates are by analyst query.

At the work station, the analyst has available a correlation support tool kit. The analyst can invoke any of the algorithms available to the automatic processing, on any subset of tracks or contacts copied to the work station. The tracks and contacts on the work station are temporary copies of the permanent tracks maintained by the automatic portion of the system. The automatic processing continues the correlation process on new data with no interference from the analysis performed at the work stations. Automatic updates to the temporary track copies are communicated to the work station to keep the analyst working with current data. The results of analyst manipulation of the temporary tracks become permanent only when the analyst specifically requests an update to the master copy of the track. The fifth principle of the architecture (decoupling analyst and automatic work rates) is addressed by this separation of analyst and automatic processing.

The data flow through the automatic processing can be viewed as having two dimensions: through algorithms and through "channels". The driving throughput performance factor in any correlation system is the number of contact-to-track comparisons which must be made to achieve a successful correlation. Both dimensions of the OBU data flow address this factor. The first dimension is the flow of data through the algorithms within the traditional correlation and tracking functions shown in Figure 4; namely, contact-to-track correlation, track update, track initiation, and track-to-track correlation.

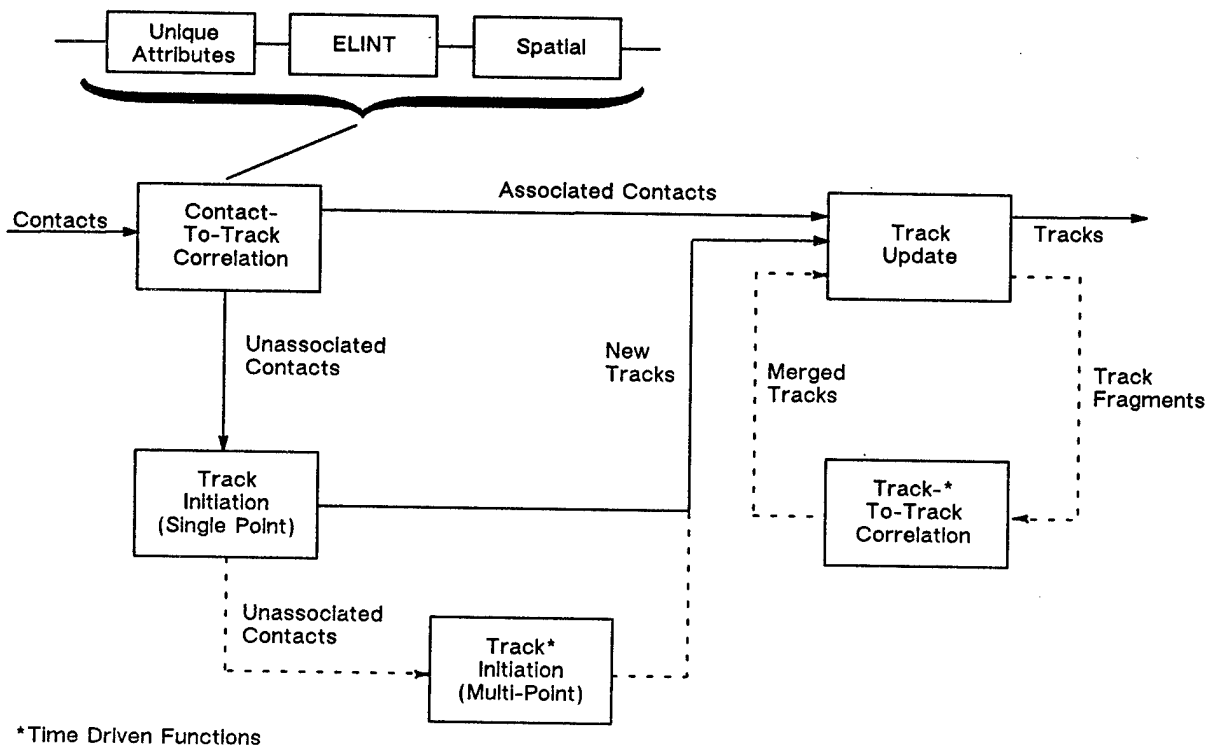


Figure 4. OBU Correlation Functions

Serial Application of Fusion Algorithms

Within the contact-to-track and track-to-track correlation functions, data is treated serially using Unique Attribute, ELINT and spatial algorithms. These algorithms are applied in order of increasing computational complexity and expected probability of correlation success. Combined with physical data consistency tests, the algorithm application order serves to "winnow", at each step in the process, the remaining track candidates to compare to the contact report. More computationally expensive algorithms are employed on fewer contact/track pairs, thereby increasing the efficiency of computer use, processing throughput and timeliness. The third ("divide and conquer" for performance) and fourth (protect against "bad" data) principles of the architecture are supported by the algorithm application order and the consistency checking performed.

The first processing performed for a contact report against a set of candidate tracks (obtained by a process called "channelization" described in the next section) is unique attribute correlation. This processing searches for matches between the contact attributes listed in Table 1 and the candidate tracks. A match of one or several attributes to one, and only one, track causes a correlation. A mismatch of any attribute (subject to the considerations discussed in the Selected Algorithms section) causes the track to be eliminated from further consideration, including ELINT and spatial processing. A match of one attribute coupled with a mismatch of another attribute, or the occurrence of matches to more than one track, causes the generation of an ambiguity. The contact cannot belong to more than one track; if unique indicators suggest it should, either the contact report is defective or the track data base contains a correctable track fragmentation. The analyst must determine which is the case.

Before a track update is performed resulting from unique attribute correlation, physical data and emitter complement consistency checking are performed. The track and contact must be geofeasible; it must be possible for the platform to traverse the necessary distance in the allotted time. There also must be no disagreement between physical data fields in the track and the contact; agreement is not required because this information is often not reported.

When unique attribute correlation is not successful, physical consistency checking is used to reduce the candidate track set prior to performing the computer resource expensive ELINT and spatial algorithms. For ELINT contacts, an allowable emitter complement table is consulted to eliminate candidate platform tracks belonging to classes that cannot carry the reported emitter.

ELINT correlation is performed using the TERESA algorithm on contacts containing emitter data from previously calibrated sources and ELINT Notations. Those ELINT contacts not qualifying for TERESA processing are treated as location data and passed to the spatial algorithms. The TERESA algorithm returns one of three results: a hard call assignment, an ambiguity, or a new emitter determination. Both hard call assignments and new emitter determinations are based purely on ELINT information, without regard to location data (previous consistency processing determined that the candidate tracks are geofeasible). When an ambiguity is generated, the system will use the spatial correlation algorithm to attempt to resolve the ambiguity. Likewise, when a new

emitter determination is made, spatial algorithms are used to attempt to find a platform track consistent with the emitter. A platform track can carry up to 8 emitters.

Spatial correlation algorithms are used if unique attribute and ELINT correlation are not successful. The MTST algorithm is being implemented in the OBU system in two phases. The current system uses the Kalman filter and the decaying speed IOU motion model for track projection. The backward smoothing (allowing more precise interpolation for out-of-sequence correlation) and aberrancy removal portions of the MTST approach are currently in development. The value of aberrancy removal, given the hierarchical nature of the OBU correlation algorithms, is discussed in the Selected Algorithms section.

Parallel Adjustment of the Fusion Process

The second dimension (through channels) to the data flow is illustrated in Figure 5. The first three principles of the architecture (flexibility, extensibility, and "divide and conquer" for performance) are accommodated through the process called "channelization." Channelization is the partitioning of incoming contact reports and tracks in the data base into subsets for processing. Each channel is defined by a query statement formed from a logical combination of contact and track attributes. Contacts are placed into a channel by comparison to the channel's query statement. The same query statement is used to automatically retrieve tracks for potential correlation to the contact. Following is an example of a query statement:

```
Channel 1: [[CAT EQ NAV] AND NATO AND MED]
          NATO: ["IC" LE CC LE "TU"]
          MED: IN RECTANGLE [3000N, 4600N,
                              00600W, 04000E]
```

Query statements can be nested as shown in this example, where the terms NATO and MED are defined by separate query statements. The example query statement will route contacts to, and retrieve tracks for, channel 1 processing that have a physical data attribute of category NAV (Naval), a reported country code that falls in a list between IC (Iceland) and TU (Turkey), and a position within a lat/long box defined by the limits in the RECTANGLE statement. In the OBU system, up to 40 query statements define contact-to-track association channels with 10 more for track-to-track association channels.

The partitioning process greatly reduces the number of tracks that must be compared to a given contact report. A poorly attributed contact (one with little information other than a position and time) must still be compared to a large number of tracks; lower priority, more general channels can be defined for processing them. The tracks are evaluated each time a new contact is placed into a channel because changes to a track since the last time the channel received data for processing may change its channel qualifications. While a contact will be placed in only one channel, a track can be put into many channels and therefore updated by many kinds of contacts. For example, consider the following four channel definitions:

```
CHAN01: [[CAT EQ NAV] AND NATO AND MED]
CHAN05: [[CAT EQ [NAV OR UNK]] AND NATO
          AND MED]
CHAN20: [[CAT EQ [NAV OR UNK]] AND [NATO
          OR [CC EQ UNK]] AND MED]
CHAN30: MED
```

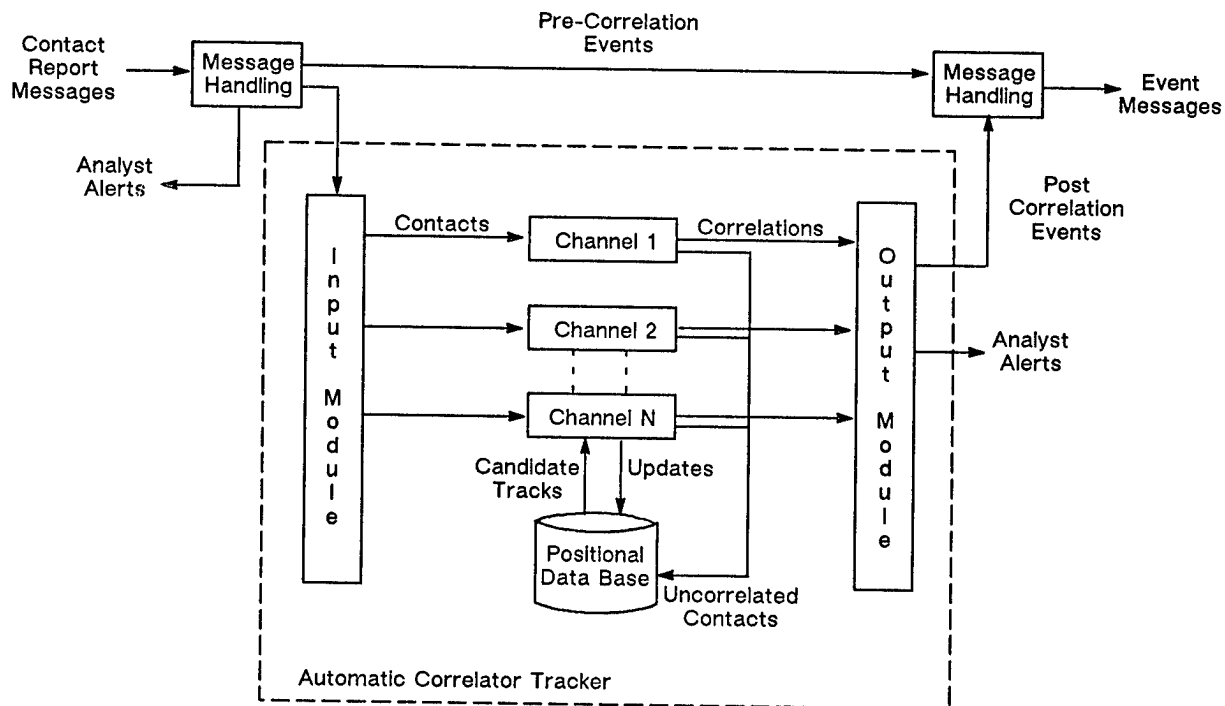


Figure 5. OBU Correlator Channelized Data Flow

This example introduces the concept of priorities to the channelization scheme; channels are allocated computer resources according to their channel number, with channel 1 having highest priority. Contacts are processed in the highest priority channel with a matching query statement. In the following discussion of routing contacts and selecting candidate tracks for the example channels, it is helpful to refer to Figure 6. In the example, all contacts reported as NATO Naval platforms in Area MED will be processed at highest priority. Reported NATO contacts, that might possibly be Naval platforms (those for which category is not reported) are processed in channel 5. These contacts are processed against all known NATO Naval tracks that are also included in channel 1 and against all NATO Unknown category tracks. Channel 20 processes contacts with unknown country (CC) and either unknown or Naval categories in Area MED against Naval or unknown category tracks from NATO or unknown countries in Area MED. Finally, channel 30 processes contacts not qualifying for higher priority channels: in the example, those with known categories other than Naval (e.g., Fishing, Merchant, etc.) or known countries other than NATO (e.g., UR) in Area MED. Note that a Greek Naval track in Area MED meets all four channel definitions; it will be retrieved and processed as a candidate against contacts in all four channels.

Channelization allows dissimilar types of tracks (air, land and wet) to coexist in the same data base, be retrieved by the same types of queries, and be processed by the same software, without increasing the number of contact-to-track comparisons per contact report processed. It allows each OSIS site to customize its processing through the logical definition and prioritization of

processing channels. Beyond the logical definition of the channels, it provides an additional degree of flexibility through tailoring the algorithms. The correlation algorithms can be made to operate differently in each channel. The three levels of algorithms (Unique Attribute, ELINT and Spatial) can be enabled or disabled by channel. Examples are to disable spatial correlation on an Air category channel or to disable unique attribute correlation in an ELINT channel (which can be defined by sensor or ELINT Notations). In addition to enabling or disabling correlation algorithms, the ELINT and spatial algorithms have sensitivity adjustments that can be used to adjust the interdependent rates of missed correlation, correct correlation, wrong correlation and ambiguous correlation. An example is setting lower spatial correlation thresholds for submarine tracks (which are infrequently observed and relatively small in number) than for naval tracks (which are more frequently observed and larger in number). Land Mass Avoidance, an intensive computational process, can also be enabled or disabled on a channel basis, enabling it for areas and categories of tracks where the benefit exceeds the cost. All of these adjustments can be made at each site as a part of customizing the fusion process to the environment.

A benefit of channelization to be exploited in the future is customizing correlation algorithms on a channel basis. For example, a stationary ELINT target tracking algorithm for correlation of land based radars is in development. This algorithm can be inserted into channels designated for processing of land category contact reports. As the need arises, further algorithm specialization can be developed and inserted into the basic channelization architecture.

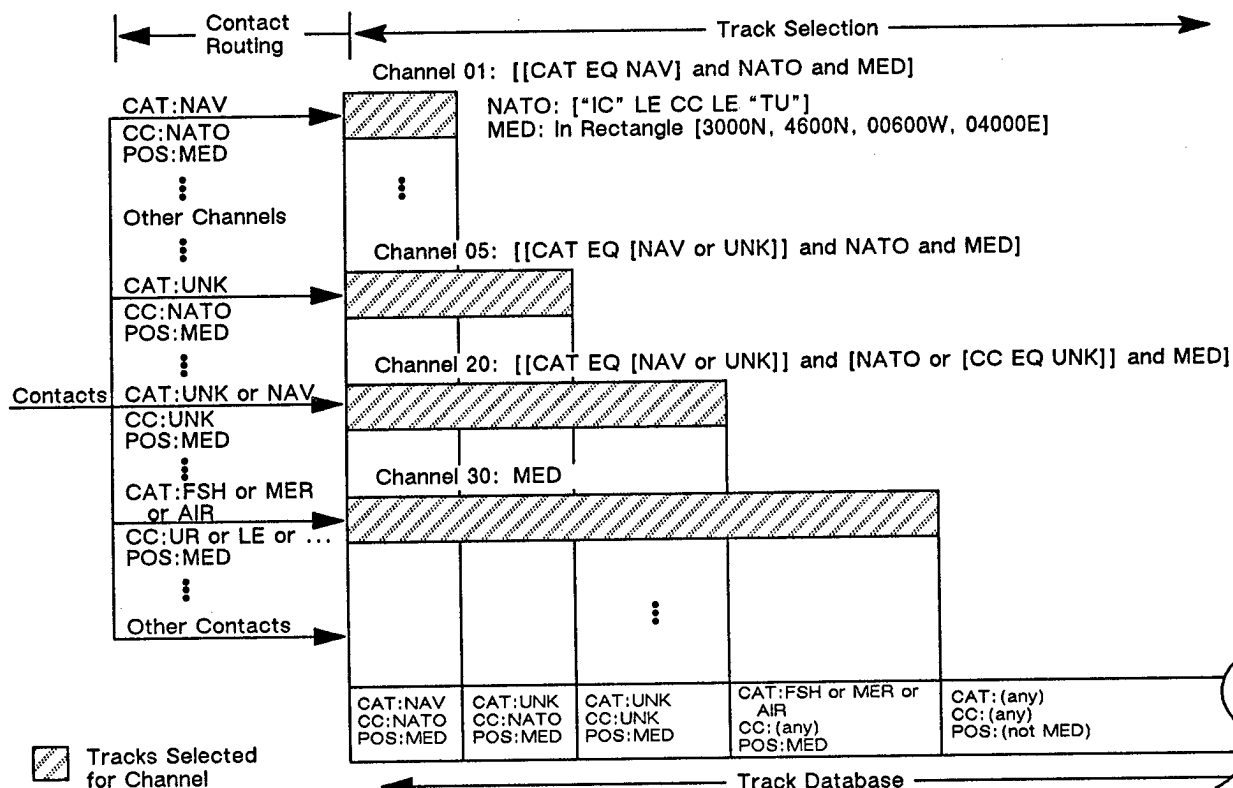


Figure 6. Channelization Reduces the Number of Contact-to-Track Comparisons

Initiation of New Tracks

The strategy used to initiate new tracks is important in a data fusion system expected to maintain a large track data base. The number of contact-to-track comparisons that must be performed is a function of the number of tracks in the data base. Even when "winnowing" and channelization are used to reduce the number of candidate tracks, indiscriminate starting of tracks may result in too many tracks for efficient operation. The analyst must deal with clutter which is also a function of the number of tracks in the data base. Starting tracks that neither grow, nor represent important events, is undesirable for both the analyst and the automatic software. On the other hand, too conservative an approach to track initiation results in lost correlation opportunities. The OBU architecture avoids the classic correlation dilemma of either always starting single point tracks, thereby cluttering the data base; or starting only multiple point tracks, thus losing timeliness and early correlation opportunity.

Figure 4 illustrates the combination of single-point and multi-point track initiation functions in the OBU system. Single-point initiation is considered for every contact that does not correlate during contact-to-track correlation. New tracks are initiated based on analyst defined "start rules." A set of rules can be applied to uncorrelated contacts in each of the 40 contact-to-track correlation channels. These rules have the same form as the channelization queries statements, for example:

```
IF [ELNOT EQ [C1234 OR C5678]] THEN
START
IF [[NOSICID NE NULL] OR [CALLSIGN NE
NULL]] THEN START
```

The first start rule initiates single point tracks from any uncorrelated contact having one of the two specified ELINT Notations. The second rule starts a track for an uncorrelated contact having a reported NOSIC ID number or Call Sign. This technique allows timely initiation of important tracks without cluttering the data base by starting a track for every uncorrelated contact.

By contrast, multi-point track initiation is performed periodically in one of 10 specially defined initiation channels. Uncorrelated contacts for the channel (which may be defined by an area, a time window and other contact attributes like ELINT Notation) are gathered and searched for potential great circle route, constant velocity multi-point tracks. This approach will be most effective in open ocean areas where movement tends to be in straight lines for a period of time. In highly congested areas, or regions in which platforms tend to maneuver, it will be less effective.

Part of the operational tailoring that the OBU architecture allows is illustrated by Figure 7. The figure shows a port with a ship maintenance area (Area 1) within it. It also shows an area off the coast where ships normally conduct exercises (Area 4). Tracking in Area 1 on either ELINT or spatial information is nearly impossible due to the erratic nature of both types of information (maneuvering in congested areas can make spatial behavior appear random; parametric distortion from maintenance and test activity can cause emitters to be inconsistent

with previous observation). The OBU system allows definition of a set of "restricted areas" where the correlation algorithms do not operate. Contact reports in these areas are not discarded, and restricted areas can be enabled or disabled on a channel by channel basis; the channelization scheme can allow unique attribute information resulting from "port watch" to be correlated while not attempting ELINT correlation in the port area.

The areas defined in Figure 7 can be used to establish a track initiation strategy for the part of the world represented. The strategy might call for not starting new tracks in Area 1 because tracking in this area is too difficult. Start rules might dictate starting single point tracks in Area 2 as indicators that a deployment is underway. Multi-point initiation might be specified for contacts in Area 3 that were not associated to tracks started in Area 2. The multi-point initiation channel definition might also exclude consideration of contacts in Area 4 because it is known that significant platform density and maneuvering are common. In addition to the multi-point initiation defined for Area 3, specific single point initiation rules (such as detection of a unique attribute like Call Sign) might also be defined.

DEVELOPMENT REALITIES

Laboratory versions of data fusion systems exist in synthetic simulated environments that can be controlled to allow concepts and algorithms to be demonstrated and proven. When developing an operational data fusion system for deployment into the field, there are additional practical realities which must be addressed. These realities can complicate implementation of the system, but to ignore them is to risk producing an algorithmically correct but operationally unacceptable system. Four of these realities are discussed below.

Fusion Algorithms Need Supporting Software

The amount of support software in data fusion systems varies greatly, depending on the environmental factors discussed in the first section of this paper. The OBU system as a "dissimilar source", "autonomous", "receptor/transmitter", "undersampled" correlation system needs more supporting software than systems having simpler environments. In Table 2, the software in the OBU system is allocated to functions which support the overall fusion mission. The Environmental Factors column indicates which OBU functions are influenced by which environmental factors. The total software represented by the functions in the table is approximately 600,000 lines of code.

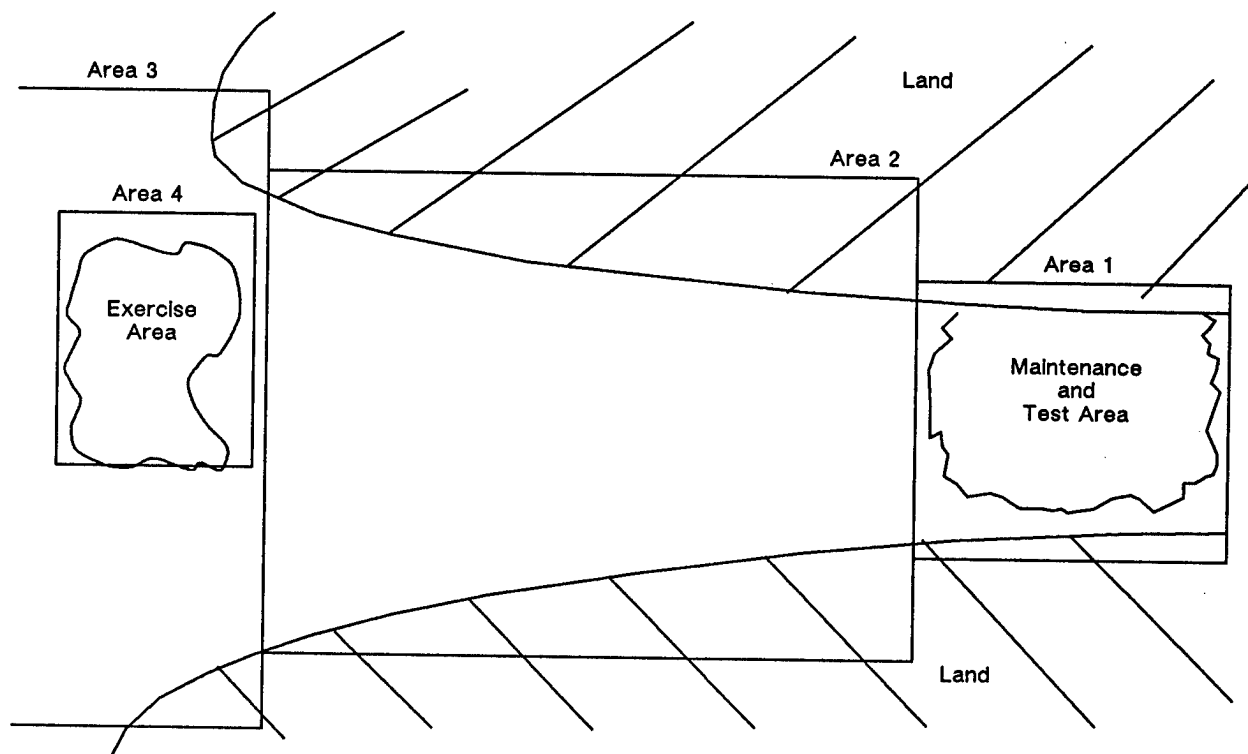


Figure 7. Definition of Operationally Oriented Regions for Initiation of Tracks

Table 2. OBU Software Breakdown By Function

Function	Percent Of OBU System Code	Environmental Factors
Correlation, tracking	7	D, A, U
Comms, Input msgs	7	D, A
Report gen, output msgs	13	D, R/T
Analysis tools, MMI	17	D, U
Data management	23	D, R/T, OC
System setup, control	33	OC

Environmental Factors:	
D	Dissimilar Source Factor
A	Autonomous Location Factor
R/T	Receptor/Transmitter Destination Factor
U	Undersampled Information Factor
OC	Operational Considerations

The correlation and tracking software is influenced by the dissimilar types of source information, the autonomous location of the system away from the sensors (allowing out-of-order receipt of information) and the undersampled nature of the data. Even with these factors, this software represents less than ten percent of the overall code in the OBU system.

The dissimilar source and autonomous nature of the OBU system requires significant communications software to receive messages from the several circuits providing data. Input message handling must translate a variety of message formats into the homogeneously structured, normalized data required by correlation algorithms. The code in the OBU system necessary to receive and condition the data prior to correlation equals that needed to perform correlation and tracking.

The Receptor/Transmitter nature of the system requires report generation, message construction and message transmission capabilities. The fusion of dissimilar security classification data complicates this software. Because of the need to provide a word processing capability to generate unformatted output reports, the amount of code to support the generating and sending of reports and messages is nearly double that needed to perform correlation and tracking.

A robust set of analysis tools and man-machine interfaces is necessary because the dissimilar sources (and the dissimilar errors they make), when coupled with the undersampled nature of the data, make completely automatic correlation unachievable. About one-sixth of the software in the system contributes to support of analysts activities.

Data management is impacted by the fusion of dissimilar security classification data from different sources combined with the external reporting nature inherent in a Receptor/Transmitter system. The existence of multiple analysts performing interactive analysis and generation of output reports for external consumers is an operational consideration which adds to the data management load. The software to manage updates to, retrievals from, and

integrity of the resulting distributed data base constitutes nearly one quarter of the system.

System setup and control software manages the local network architecture, which was selected largely based on operational considerations: throughput and timing requirements forcing separation of the analyst from the automatic data flow, and system availability requirements. A third of the OBU software falls into this category, which also includes software necessary to control operation of the correlation process.

Real Systems Must Process Real Data

Information coming into an autonomous data fusion system passes through a potentially distorting filter called a message format. Even though formats are generally standardized, reporting conventions can add nuances to individual data items that must be considered. A system receiving a variety of message formats must accommodate any bias between sources in the use of the same or similar message fields.

Correlation algorithms based on unique attribute data are particularly sensitive to imprecise understanding and interpretation of specific message format fields. This sensitivity is due to the deterministic matching and mismatching of attribute fields inherent in these algorithms. Differing conventions between sources reporting similar data must be understood and accommodated in the processing.

Parametric data can be distorted through the message formats by truncation. An ELINT sensor might send out contact reports in several different formats. Not all formats allow the same number of digits of precision in their parametric fields. A correlation site may need to choose between using more timely information with n digits of accuracy or later information with $n+1$ digits of accuracy. The effect on the mathematics of the correlation process is not clear, especially if n and $n+1$ digit samples are combined together in statistical calculations.

Designers often rely on the existence of data elements defined in message format specifications. In fact, many fields may be reported rarely or used only by a subset of the sources. Only through analysis of representative message traffic from all the potential sources can the frequency and reliability of specific data elements be determined. Even data elements which are specified as "mandatory fields" in message format specifications are often reported as "/-/" ; the field was filled, meeting the format requirements, but no information was provided.

The need to obtain "representative real data" to support the design and testing of a data fusion system is compelling. The difficulty in so doing can be significant. Security considerations, data variations by theater, and the influence of speculations and myths about the data are factors. Only by consulting data experts in the field can many of the subtleties of the data be discovered. Yet, consultation may only be fruitful after prior study of sample real messages. Analysts in the field are generally more familiar with observed message characteristics than with message specifications.

Another aspect of real data is that the sensor reporting world is dynamic. Early software development must protect itself from change by tightly controlling the assumptions about message formats and using artificial or simulated data to test interfaces. This test data can be effective if generated from an understanding obtained from analysis of messages. This approach alone is often

not enough to ensure success. Samples of real messages must be periodically analyzed to reaffirm or challenge earlier assumptions. Six months is a recommended interval for reexamining the content of the messages in the field.

Security and Recovery are Real and Difficult Problems

In a speech on Navy fusion requirements at the 1987 Tri-Service Data Fusion Symposium, RADM K. L. Carlsen stated: "These security barriers are a fact of life; we need to recognize that fact in our thinking about fusion and accommodate it early in the design of fusion procedures and equipment" (Reference 7). The OBU system was developed with this goal in mind. An additional operational constraint was the need to recover from hardware, power or software failures without losing either the accumulated knowledge represented in the track data base or the contact reports currently in process.

Security and recovery are not usually important requirements in the development of test-bed or proof-of-concept prototype systems. In any deployed production fusion system, however, these issues must be successfully and efficiently addressed. "Successfully" addressing these issues from a functional point of view is a matter of software engineering and was achieved during the OBU development in a straightforward manner. The "efficiency" aspect was more difficult and took considerable effort; both security and recovery exact a high price in computer resources. The challenge for any large scale data fusion system development is to meet required functionality at an acceptable cost in performance.

Support Data Bases Cannot be Ignored

Many data fusion systems use support or collateral data bases in their processing or to assist the analyst. These data bases can be as varied as land mass avoidance, shipping lane definitions, platform emitter complement files, or emitter mode definitions. These data bases must be defined and sized early in the development. A strategy for populating, validating and providing life-cycle support for these data bases also must be established. Too often, only sample data bases to support testing are developed; the development assumes that the "real" data base will be generated by operational personnel after the system is deployed. As a result, data fusion systems have been fielded which do not live up to their potential because necessary support data bases are too difficult or too time consuming for busy site analysts to populate. Just as important is the ability to maintain support data bases once in the field. Without effective tools, modification of support data bases may be beyond the capability or inclination of a data fusion site. Out of date or incorrect support data bases could degrade a system's utility. In some cases, the type of information in a data base makes it necessary to obtain new releases periodically from a central location. In this situation, a method of installing new data base releases which accommodates site specific customizations must be considered.

DEVELOPMENT CONSEQUENCES

How the Development Realities discussed above affected the realization of the OBU system is illustrated in the following examples. These realities not only influ-

enced this development but are applicable to future large scale fusion system developments.

System Support for Data Fusion

A major consequence of the environment factors on the OBU development was the need to develop considerably more software than is evident from the fusion algorithms. Data handling, data management, analyst tools and system control became major aspects of the development effort. At times during the development, the computer science aspects of the system became as important as the algorithmic aspects. Once such example follows.

Accommodating the inherent mismatch between analyst working speeds and automatic processing speeds while minimizing mutual interference proved a challenging software task. One of the main technical dilemmas was how to "lock" tracks to avoid simultaneous update. This is a classic data management issue, but in this case there were additional considerations. If the analyst was allowed to lock tracks during an entire period of analyzing and manipulating them, the automatic software would fall behind in its work because a correlation channel cannot run unless all of the tracks in that channel are available. Systems with lower throughput requirements or single analysts can often afford to lock a track or the entire data base while an analyst manipulates a track: the OBU system cannot.

To ensure that the processing resources invested in performing an automatic correlation are not wasted, the automatic software locks all tracks in a channel before beginning correlation in that channel. Thus, there is a guarantee that a successful correlation will result in update of a track, and that the track did not change since the start of the correlation process. The analyst must also lock tracks, but only when he is ready to do an update. During active periods, the automatic software can generate a high demand for certain tracks and many resulting requests to lock these tracks. These are often the same tracks needing analyst resolution of ambiguities. The analyst cannot wait indefinitely to perform an update; at some point he must receive a "busy, try later" response. However, if he is unsuccessful in performing an update for too long, the automatic software may update the same track, making the analyst's work invalid. The solution to providing timely analyst access to tracks without impeding automatic processing is to institute an analyst-biased reservation system. In this way, the automatic processing can complete current processing but the analyst will always get next access to the track for update, regardless of how many requests for the track the automatic system has pending. The analyst update will only lock the track momentarily and then the automatic system can continue.

Another aspect of synchronizing the analyst, who works slowly compared to a computer, and the automatic correlation, which works quickly, is how to organize and control the flow of data from the computer to the analyst. The analyst may fall behind, especially during heavy data load periods. Real data does not arrive at a uniform rate; there are distinct peaks and valleys. During peaks, the analyst must be able to do important work first and defer less important work. The OBU system organizes analyst work into "work list" queues. Items can be put into queues by alert filters within the system and by the correlator generating ambiguities. To allow the analyst selectivity in the ordering of work, each of the channels in the correlator is capable of routing its ambiguities to separate work queues. By planning the setup

and allocation of these queues, the analyst can use this division of work to help establish priorities. Within a given work queue, a two line summary of the work item allows an initial assessment of the urgency of the particular work item.

Large systems have ancillary functions that may be considered less important during development, and therefore receive less attention, than the main functions of the system. An ancillary function in the OBU system is the logging of information and status. Many types of information are logged: incoming and outgoing messages, significant events associated with contacts, security relevant events, significant software processing events, and system alerts. There was much discussion during the development of the system of what data items to log and where to log them, but the concept of how the results of logging were to be used was not fully developed. As a result, the logging is thorough, but the extraction of information from the logs to perform specific analysis is sometimes cumbersome. An effective logging system needs to emphasize the information to be extracted from the logs and the goals of extracting this information, before becoming involved with the detailed definition of what is to be logged.

Real Data

Use of unique attribute data for correlation can cause correlation anomalies resulting from ambiguous interpretation of message format fields. For example, in the initial OBU system, the unique attribute correlation algorithm deals with two types of sensor supplied track numbers: the Precor (Precorrelated) Track Number and the Raid Number. The Precor Track Number is taken to indicate transmission of an entire track at one time, while the Raid Number indicates an update to a previously reported track. The OBU correlation processing is different for these two unique attributes; if a whole track is received, there is no need to search for previous points belonging to the track in the data base. Unfortunately, some sensors can report track segments (several points for a track in a single message); the handling of a track segment encompasses the rules established for both Precor Track Number and Raid Number. The first receipt of a segment must be treated like a Precor Track Number (put all the points in the data base as a new track), whereas subsequent segments should be treated like Raid Numbers (find the existing track and extend it). The message processing software extracting the "track number" has no reliable way of assessing which kind of track number is being received. In the future this will be resolved by combining the track numbers into a single unique attribute, relieving the message handling software from making an unreliable decision. This simplification in message interpretation must be accompanied by additional processing in the correlation software. Upon receipt of a track number, the track data base must be queried to determine existence or nonexistence of a track matching the number received.

Another example deals with the reporting of (Radio) Call Sign, which is also a unique attribute in the OBU system. In general, a given Call Sign can only be used by a single platform. However, certain conventions have evolved in the reporting world which must be recognized. For example, the Call Sign "NC" means "No Call" or the equivalent of "Unknown." A system correlating on Call Sign, without recognition of this special case, will attempt to correlate all contacts having Call Sign NC into a single track, generating incorrect results. Similarly, certain Call Sign values are used as a special convention to report

collateral information other than Radio Call Sign. Knowledge of the existence and values of these special conventions must be built into the unique attribute correlation process.

The OBU display software initially used the value of the reported "Force Code" to determine the symbols and colors to display for a track. The goal of this coding is to make the displays more readable by the analyst. However, analyzing message data from many sensors indicated that "Force Code" is rarely reported, even though the field exists in the message formats. As a result, appropriate track symbols and colors were not generated. Most sensors prefer to report country code (flag) and category. Force code will be derived from this information in the OBU system to allow correct display of colors and symbols.

The OBU system was designed with the recognition that message format interpretation difficulties are unavoidable, especially in a large system processing many formats from many sources. An ability to modify message parsing rules without changing software has helped the system adapt to the real data encountered in the field.

The multi-source nature of the OBU system requires information manipulation prior to the correlation process. The need to access contact report information and track information by software throughout the system led to the definition of a standard data structure for contact reports, called an ICR (Internal Contact Report), and one for the aggregate track information, called a track header. These data structures are used by correlation software, display software, data management software, report generation software and by the various analyst tools. A single format for dealing with information, independent of the sensor or message format, is convenient during software development. It results in easy to produce and understand code, a definite goal in modern large-scale systems. It also appears efficient from a performance point of view, because once information is placed into the initial structure, further data transformations prior to use are not required. However, due to the wide variety of sensors and the dissimilar information they report, these structures are significant in size (ICR = 1024 bytes, track header = 2560 bytes) in the OBU system. They are often very sparsely populated with information because no one sensor reports (or platform has) all of the attributes allotted space. As a result, the system often stores and moves largely empty data structures. Simplicity and clarity in the implementation extracts a subtle penalty in storage and performance. In the area of performance, the gain in avoiding transforming the data structure each time it is used (e.g., by packing and unpacking) is at least partially offset by the cost of moving larger structures. From a simplicity point of view, the single format is a definite advantage; from a performance point of view, the advantage depends upon how much empty space must be moved through the system and the cost of this movement. This is a trade-off that "dissimilar source" data fusion systems must make.

Testing to Approximate Real Data

Testing is a subject related to the characteristics of real data. Initial testing is usually performed with artificially constructed data that can conveniently produce needed scenarios. Over time, the test data must evolve to become a better representation of the real world. Testing of a data fusion system requires a number of tools. "Ground truth" data must be generated by simulation and used to test algorithms. During early integra-

tion of the OBU system, debugging efforts sometimes showed that the software worked properly but the test data was defective. A need was realized for additional tools to perform quality assurance checks on simulation generated test data. As test scenarios reached multiple thousands of contacts, independent verification of the correctness of the data became more important. Using unproven data to test unproven software makes problem resolution less timely. An engineering challenge recognized early in the OBU development was management and quality assurance of the test data to achieve variety without losing control.

During early testing, the correlation software was separated from the message handling software. Simulated contact data was fed directly to the correlation software. The simulation placed the data into Internal Contact Report (ICR) format. After a time, the correlation software worked properly and it appeared to be a straightforward matter to combine the correlation and message handling software. Several unexpected practical details made this transition take longer than expected. First, the data simulation software now had to translate the contact report information into various message formats. A way had to be found to store "ground truth track number" in each message to allow evaluation of correlation correctness by test tools. This had to be done without influencing the message handling or correlation processing and invalidating the tests. After this detail was resolved, a second impediment had to be overcome; when the message handling software parsed the various message formats into ICR format and passed the results to the correlation software, the correlation answers were sometimes different than previously. Differences arose from how the simulation software populated the fields in a message, from how the message handling software parsed the message, from how default values for missing message fields were handled, and from how message field translations were performed. A correlation subsystem that can develop proper tracks is not completely checked out until it can develop tracks in cooperation with a message handling subsystem. Much of the work in accomplishing this transition lies in the very low level practical details of adapting test tools and determining the correct construction and parsing of messages. Many detailed message construction or parsing issues do not surface until the message handling and correlation software begin to work together.

To thoroughly test and evaluate a complex correlation system like the OBU system, a large amount of test data must be processed. This data is generally processed in batch runs and then results are analyzed. The underlying philosophy in the OBU correlation process is to rely on the analyst to repair inconsistencies in the data and to make decisions the machine cannot make because candidates are too similar to distinguish. "Ambiguities" deferred to the analyst tend to cascade over time if not resolved. Runs of many thousand contacts without working off ambiguities can result in a large percentage of the later contacts generating ambiguities. This occurs because earlier ambiguities have not been resolved; if left in the data base, contacts similar to existing tracks, not correlated due to ambiguous situations, tend to start new tracks. The result is yet more tracks similar to earlier tracks, with the potential to cause future ambiguities. On the other hand, providing reasonable and repeatable manual ambiguity resolution during test runs without biasing the test outcome is a challenge in a software development environment.

Security and Recovery

Security information must be fused by a multiple source correlation system that automatically reports results over communications circuits (receptor/transmitter category fusion systems). Classifications, code words and releasability instructions must be fused. As a result, over time, the information in a track will evolve to the highest levels of classification, as each contact added to the track adds security information. The track information becomes unavailable for reporting to consumers with lower level clearances and the ability of the system to effectively discharge its mission can be diminished. A major dividing line between consumer clearance levels is whether they are limited to receiving GENSER (General Service) level information only or whether they can also receive SCI level information. In performing multi-source fusion, the OBU system receives both categories of information and must support both categories of consumer. In order to meet this requirement, the system keeps two sets of books on the track data elements which are used for insertion in outgoing event-by-event messages. For a given element, there is a basic track copy that is the fused result of all contacts in the track, and a GENSER-level copy that is the fused result of only those contacts received with GENSER security classification. Based on which consumer is to receive information, the appropriate copy of the track is used.

Details of the code can become important in fusing security levels. Consider an example where a track contains a radio call sign of ABCD. This call sign and all of the remaining data in the GENSER track copy came from contact reports classified "confidential". Another contact is received that is classified "secret" and contains the same call sign value "ABCD". If the software does not detect that the call sign value "ABCD" already exists in the track, and instead overwrites the existing value "ABCD" with a new copy of the same value "ABCD", the security level of the track must be upgraded to "secret" because the information was derived from a secret message. On the other hand, if the code realizes that it already has the call sign value ABCD in the track, and does not update the information, the level of classification can remain "confidential".

Security and data base recovery requirements serve to exacerbate the complexity of the throughput performance estimation effort. When added to the inefficiencies inherent in productivity enhancing high level languages and vendor supplied system software services like file management, system throughput and processing timeliness can no longer reliably be derived from algorithm complexity. The technique of determining the number of mathematical operations or lines of code an algorithm requires and adding 15-25% for operating system overhead is no longer valid in modern complex systems; if relied upon it can result in unexpected performance shortfalls. More sophisticated methods of analysis, measurement and finally optimization must be applied in a system that must meet particular levels of timing performance. The OBU software was subjected to significant scrutiny and rework during early integration to achieve required levels of performance in the face of stringent security and recovery requirements.

Either multilevel security or recovery implementation, if not carefully designed, can reduce a computer system to 25% or less of its expected performance; their combined effect can leave only 10% of the computer for application processing. With performance optimization, the effect of either can be reduced to the point where the machine will perform at 75% of expected performance.

Thus, the compound effect of multilevel security and recovery can still reduce the application level performance of the computer system to 50% of the performance possible without these factors. These numbers represent estimates from examining the OBU system and several other similar large systems. Actual determination of the computer cost of security and recovery is not practical because both features must be designed into the very basics of the architecture and cannot be separately measured.

Although the magnitude of performance impact from security and recovery is similar, the underlying causes are different. Security architectures require separation of processing of data with different security levels. This separation mandates movement of information within the system coupled with the cleansing (by zeroing) of storage space previously occupied by higher classification information. The continual movement of data and zeroing of memory effectively empties the high speed cache memory used in the computers which host the majority of data fusion systems. These computers depend on successful retention of frequently used information in cache memory to achieve advertised levels of performance. The activities required to build secure systems work counter to modern computer architectures.

The performance impact of recovery requirements is from a different cause. As a part of data base recovery, data base integrity must be considered. A data base update to reflect assignment of a contact to a track requires the modification and/or creation of several data base entries along with linkages between them. If a system failure occurs in the middle of a string of such transactions, the data base is left in an internally inconsistent state. In recovering from a failure, the incomplete sequence of transactions must either be completed or rolled back to the beginning to start over. In order to provide a path to traverse to take the data base back to a consistent state, information must be saved at various points during processing. This information must be stored (called "journaling") on a disk to allow its recovery after a failure. In a distributed data base system with a high transaction rate, and multiple updaters of the data, the resource cost of journaling can be significant.

Support Databases

The OBU Land Mass Avoidance (LMA) data base can be used to highlight the issues of developing operationally useful support data bases. The data base consists of two parts, a land/water/shore map and a "points and line segment" set of possible routes. Personnel in the field will have better knowledge of the shipping lanes traversed by platforms of interest than do software developers, but personnel in the field may not have time to build data bases. Therefore, the OBU LMA data base was computer generated by connecting points derived from standard map data bases. Obvious shipping lanes like the Panama Canal were entered manually. The two part land mass avoidance data base now consists of several hundred thousand points. A testing program was initiated to detect and repair anomalies in the data and algorithms. Unfortunately, it was found that testing all paths would take over a CPU-year. A random testing program was then initiated. After several months of testing and repair, the data base anomaly rate was reduced to 7 or 8 per million data base excursions. The errors continue to be corrected, but at a testing cost much greater than anticipated.

The deployed data base gives the site an immediate ability to use land mass avoidance, but because it was generated from maps, it does not account for the subtleties of platform movement in specific theaters of activity. In those circumstances, the system provides the analyst a tool for the creation of additional routes. In this way, the data base can be customized to the needs of the site. On the other hand, if the site has no time to modify the data base, there is in place a data set that provides a basic level of functionality.

FUTURE DIRECTIONS AND POSSIBILITIES

Directions

The current deployed version of the OBU system has been optimized for the tracking of "wet" naval targets. An on-going development effort will result in customization for efficient correlation of aircraft and fixed land-based targets. The architecture described in this paper will remain. Enhancement will be in the areas of system responsiveness to analyst command, algorithm differentiation, and data structure definition.

The modifications planned for aircraft target tracking are largely not algorithmic but deal with increasing system responsiveness to the analyst. The current unique attribute and ELINT algorithms will be applied to aircraft platforms. In its initial orientation toward naval targets, the present version of the OBU system places emphasis on maintaining integrity and consistency of tracks, their contacts, and the updates. This is done at the cost of decreased analyst responsiveness, an appropriate trade-off in an environment that requires maintenance of long time-span, low update rate tracks. In the air environment, the update rate of individual tracks is much greater. As a result, the system must be more responsive to analyst commands. This emphasis on speed is offset by the fact that individual tracks are much shorter in duration, so long term track integrity is not as important. Trade-offs that were made in favor of long term track integrity are being revisited for air tracking with the goal of making the system more responsive to analyst command. Indications are that some of the techniques being considered for air tracks will also apply to naval tracks, thereby increasing overall system responsiveness. Optimization of the performance of commands that air analysts use frequently with prototype air processors is also being investigated. These optimizations will benefit all analysts on the system.

The advent of fixed-site land target tracking will result in introduction of additional correlation algorithms into the system. These algorithms will operate within the currently defined channelization structure on specific channels designated to process land data. Unique attribute algorithms will be enhanced to include land target unique attributes. A combined ELINT/spatial correlation scoring technique, as opposed to the hierarchical ELINT followed by spatial scoring scheme used currently, is under consideration.

The common track data structure may need modification to provide a specific variation for land tracks. Initial indications are that combining all of the necessary information into a common data structure may make the structure too large for efficient movement within the system. The land correlation problem also has a fundamental difference from the naval problem that can be exploited: a large number of targets with a low average update rate per target. This increases the probability of retrieving and considering a large number of tracks for

each contact. A dynamic channelization scheme is being investigated to use the fact that fixed land sites don't move, or if they do, not very far or very fast. Dynamic channelization performs the geofeasibility test as part of the basic channel track query, not retrieving tracks more than a certain distance from the contact.

Possibilities

Introduction of additional algorithms and capabilities beyond those currently planned will await field evaluation of the current OBU data fusion approach. The following paragraphs speculate on potential directions for the future.

Knowledge based system concepts and techniques might be applied to analyst ambiguity resolution. An intelligent analyst work station could be made capable of monitoring and classifying the resolution of ambiguities by an analyst. From this monitoring, the "ambiguity resolver" could build the rules and use patterns for various ambiguity resolution options available to the analyst. The goal would be to collect enough understanding of ambiguity resolution patterns to begin "suggesting" to the analyst, upon presentation of an ambiguity, the actions typically taken in the past (e.g., "70% of the time this category of ambiguity has been resolved in favor of the unique attribute data in the ELINT report"). The system could be trained by an experienced senior analyst; his pattern of decision making can then be made available to a less experienced analyst who works a different shift.

New ELINT approaches like DUET, may be considered if they are demonstrated to offer superior performance to TERESA at acceptable computer resource cost. Extension of TERESA to process additional ELINT parameters and to handle data from multiple sensors will probably be necessary at some point. Multiple hypothesis algorithms would require a restructuring of internal data management approaches and would be a candidate for implementation only under the following conditions: 1) if the ambiguity rate in the OBU system cannot be adjusted to acceptable levels, 2) if a better level of correlation performance can be demonstrated, and 3) if the reservations expressed in the algorithm section of this paper can be overcome. Performing a probabilistic position update according to the correlation scores of the several tracks involved in a spatial contact-to-track ambiguity is an untried heuristic approach that falls between single and multiple hypothesis techniques. A theoretical basis for such an approach would need to be developed before it could be seriously considered.

It will be necessary to address methods of integrating the capabilities of prototype systems currently operational at the OSIS sites to produce a single analyst view of the ocean surveillance picture. Functional interfacing to prototypes conforming to specified interface standards is possible. The interfaced systems would be subject to scrutiny to ensure operational and security integrity. As key features of today's prototypes are determined to be necessary in performing the ocean surveillance function, they can be introduced into the full scale engineering development process. Incorporation of these features into the OBU system as the site data fusion hub will likely be paralleled by the appearance of new prototype systems, addressing the continually changing nature of the correlation environment and the fusion site personnel's desire to experiment with new functions.

The OBU system bases correlation decisions on unique attribute, ELINT and location information. Since

each consists of dissimilar types of information, during the design of the OBU fusion approach it was felt that there was little theoretical basis to perform automatic correlation based upon a combined "goodness of fit" encompassing all of the information. The result was the implementation of the hierarchical set of correlation algorithms discussed in this paper. In this hierarchy, a correlation decision (including the decision to inform the analyst of an ambiguity) can be reached at any of the three levels. Lower levels of the hierarchy are not consulted if the computations at any level produces a sufficiently good correlation. Similarly, once a lower level is reached in the correlation hierarchy, information from upper levels is not used in attempting to make the correlation. The result is less than optimal use of the available intelligence data in reaching decisions. A decision algorithm which makes use of all three types of information to arrive at an optimal correlation decision might be developed. In particular, techniques for computing a combined "goodness-of-fit" for ELINT and spatial characteristics are a definite possibility. Additionally, provisions for site defined unique attribute "confidence factors" in specific situations (e.g., sensor 1 makes a correct NOSIC ID assignment 60% of the time for ELNOT xxxxx) might be investigated. Currently all unique attribute data is treated as being true - independent of the source, correlation situation or availability of supporting information. Applying probabilistic confidence measures to unique attribute assignment could allow integration of the unique attribute level of the correlation hierarchy with the parametric and spatial attribute levels of correlation.

Correlation Engines

Many of the ideas listed as "possibilities" above as well as new algorithms coming from the research community may well improve the data fusion capabilities of the OBU system. The basic system architecture will support significant growth in the hardware, software and algorithm dimensions. On the other hand, it may prove difficult to provide the machine power necessary to accommodate increased input data rates that might occur in the intermediate term and also accommodate newer correlation approaches (which always seem to require more computation) without degrading the timeliness characteristics of the system. In addition, as input rates increase, the number of ambiguities the analyst must handle may also tend to increase. More thorough treatment of all the information available or use of multiple hypothesis algorithms might be able to decrease ambiguity rates. Once again, more processing power will be necessary.

There are two ways to address the potential need for more processing power in the future. One is to wait for the newer and faster general purpose computers which appear every few years. The second is to consider development of a correlation "engine". Such an engine could be built from ASIC (Application Specific Integrated Circuit) technology or programmable high performance microcomputer chips. By allowing such a device read-only access to the OBU track database and treating its output as a recommendation to the OBU computer to perform a track update (add contact A to track B), such a device could be incorporated into the OBU system security and recovery architectures. This engine could provide added computing power optimized for correlation processing and would allow consideration of more complex correlation algorithms while performing more contact-to-track comparisons per unit of time and achieving greater throughput. This approach could also provide a means of adding more sophisticated algorithms to the OBU system while maintaining the strengths of the cur-

rent system. Among these strengths are the operational considerations accommodated in the basic architecture: timeliness and throughput performance, robustness of recovery and database integrity, strict and accreditable security features, extensive communication connectivity, a wealth of support tools provided to the analyst, and a distributed architecture which supports multiple analysts without interfering with the speed of automatic processing.

External Interfaces

As the OBU system takes its place as the hub of the Navy's ocean surveillance function, increased emphasis will be placed on formalizing a distributed data base concept capable of handling both ashore and afloat systems and accommodating prototypes. This effort will be required to handle data flow to and from afloat tactical systems while maintaining internal data base integrity. Considerable experience with the current system, real data and better understanding of site specific needs will be required to permit definition of a structure that meets the performance and information needs of both tactical and intelligence users.

REFERENCES

1. Kramer, G. Fred, "A Survey of New Techniques for Processing Electronics Intelligence (U)," 1987 Tri-Service Data Fusion Symposium Technical Proceedings, Volume II, pp. 1-17, Secret Paper.
2. Askin, K., Schultz, A., Osgood, C.F., Hillson, R., "Measures of Performance as Applied to TERESA and NEAT Algorithms (U)," NRL Report 8826, 20 June 1984, Secret Report.
3. NOSC, DUET (Developmental Unified ELINT Tracker) Functional Specification (U), 1 May 1983, Secret Report.
4. NAVELEX, Over-the-Horizon/Detection Classification and Targeting System Level Specification Ship Tracking Algorithm, November 1984, PDE-120-S-00454 Rev A.
5. Mitzel, G.E., Barnett, P.G., Keuhne, B.E., Sommerer, S., "Wide-Area Correlation and Tracking of Surface Ships Using Multiple Sensors," Johns Hopkins APL Technical Digest, Volume 5, Number 1, 1984, pp. 28-36.
6. Barnett, P. G., Chrysostomou, A. K., Sommerer, S., "Practical Aspects of Multiple-hypothesis Correlation (U)," 1987 Tri-Service Data Fusion Symposium Technical Proceedings, Volume II, pp. 18-29, Secret Paper.
7. Carlsen, RADM Kenneth L., USN, "Navy Requirements," 1987 Tri-Service Data Fusion Symposium Technical Proceedings, Volume I, pp. 39-80.

THIS PAGE LEFT BLANK INTENTIONALLY

REAL-TIME AUTOMATED TRACK MANAGEMENT

Susan J. Feldman

HUGHES AIRCRAFT COMPANY, INC.
P.O. Box 3310
Fullerton, CA 92634

The Advanced Combat Direction System (ACDS) Block 1, intended for all United States Navy non-Aegis cruisers and carriers, brings to the fleet significant new capabilities for automated, integrated, real-time track management. At the sensor source level, making use of government furnished information and operator-entered libraries, ACDS interprets and evaluates data from both cooperative and non-cooperative tracks. At a higher level, ACDS automatically associates tracks and then fuses classification data derived from a variety of sources both organic and external to the ship. The resulting track identification drives threat evaluation, is displayed to Combat Information Center tactical decision-makers, and is transmitted over tactical data links (including TADIL J) to other Battle Force members.

Sitting at the hub of a distributed information system, ACDS must interface both with existing systems and with new systems, many of which are in concurrent development. Each of them has its peculiar point of view for measuring and evaluating tracks and its own taxonomic language for classifying the tracks based on observed characteristics and user requirements. Clearly interoperability has been a major concern in the development of ACDS Block 1 and many related practical problems have had to be solved including taxonomies, libraries, the man-machine interface, and software implementation. Additionally, new algorithms have had to be developed to deliver required performance in the areas of track correlation and Electronic Support Measures (ESM) data integration. This paper addresses the ACDS approach to solving these problems and provides performance modeling results.

REAL-TIME AUTOMATED TRACK MANAGEMENT

Susan J. Feldman

HUGHES AIRCRAFT COMPANY, INC.
P.O. Box 3310
Fullerton, CA 92634

INTRODUCTION

The commander of a Battle Force or a Navy surface vessel in the next decade will be faced with a busy threat environment consisting of many tracks, arriving quickly and distributed across a vast surveillance region. Modern shipboard computers and Link 16 make possible the processing and communication of the high volume of associated data. But without automated real-time track management and decision support processing to reduce the data into usable information the commander and his Combat Information Center staff will be overwhelmed. The Advanced Combat Direction System (ACDS) Block 1 is being developed to meet this need for automated track management. ACDS Block 1 will be installed on U.S. Navy carriers and other large non-AEGIS surface ships during the 1990's. Block 1's very ambitious requirements, including the automated fusion of all available information for use in the real-time tactical environment, will make it the Navy's operational state-of-the-art.

PURPOSE

The purpose of this paper is to provide an overview of real-time automated track management and data fusion as they will be implemented in ACDS Block 1. The paper will address requirements, design and techniques used, performance, and related issues. The reader is invited to apply to the Naval Ocean Systems Command (NOSC) for details beyond the unclassified security level of this paper.

ACDS BLOCK 1 OVERVIEW

ACDS Block 1 will replace the Naval Tactical Data System (NTDS) on carriers and non-AEGIS cruisers (and potentially other large surface ships) with new software that will give the system expanded surveillance, command support, and weapon control plus multi-link capability. ACDS is being designed specifically for TADIL J, the protocol for the new tactical data link, Link 16.

ACDS Block 1 is required to combine real-time and non-real-time track data from all available data sources so that identified tracks will be available for immediate display and threat processing. This applies to all tracks, both cooperative and non-cooperative, in the Navy dispersed battle force surveillance volume, illustrated in Figure 1.

The ACDS track management function correlates remote and local tracks from many sources, including radar and Electronic Support Measures (ESM), and automatically identifies the combined track. Red, blue, and neutral tracks may be classified at a level of detail that is useful for command decisions and is consistent with STANAG 1241, the basis for the TADIL J taxonomy. Data link protocols for ID precedence and difference resolution are followed. The operator's principal role in track management is to control the system and act as an arbitrator when conflict situations arise.

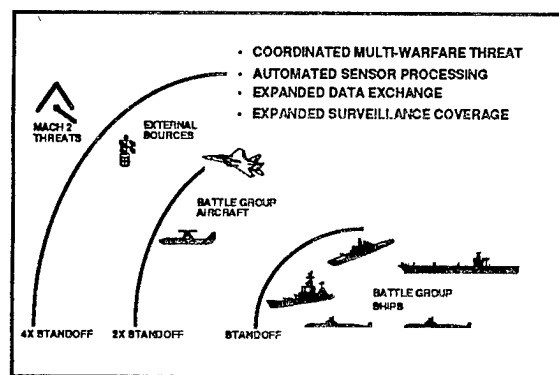


Figure 1: ACDS Block 1 Operating Environment

ACDS is hosted in standard Navy computers (AN/UYK-43) using the Navy standard high order computer language (CMS-2) within a tactical real-time operational frame. These requirements and the extension of the surveillance region to be included within the ACDS track stores dictate an implementation approach that makes efficient use of computer memory and processing.

BLOCK 1 ARCHITECTURE

An overview of the Block 1 architecture is illustrated in Figure 2. ACDS interfaces directly with the ship's sensor/trackers, weapons, and also with other systems whose principal concern is non-real-time track processing. ACDS interfaces with other battle force units, via the Command and Control Processor (C2P), over Links 4, 11, and 16. Support is provided for approximately 25 operator modes, depending on the platform.

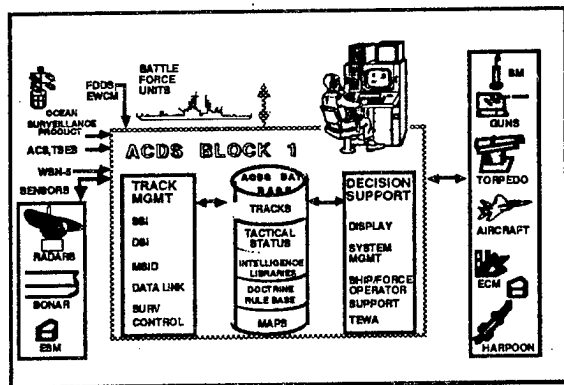


Figure 2: ACDS Block 1 Architecture

The Block 1 program is housed in two AN/UYK-43 computers, one the Track Management Processor (TMP) and the other the Decision Support Processor (DSP). The TMP is essentially a data fusion box; the major technical challenges for the TMP were: development and validation of data fusion algorithms that predictably meet quantifiable performance requirements; and (b) software techniques to manage the computational load. The DSP is essentially an expert system driven by operator-definable rules; the major technical challenges for the DSP have been: (a) devising a man-machine interface that puts the operator in control of the automaticity and presents him with complex information in an understandable form; and (b) software techniques for applying large numbers of rules to large numbers of tracks in real-time. (In this way ACDS Block 1 has incorporated into standard Navy language and computers many of the features commonly associated with artificial intelligence (AI) and specialized AI machines and languages.) The ACDS data base consists of dynamic track data and tactical status files, the operational doctrine rule-base, and government furnished libraries and maps. The TMP produces and updates the track record data which is then used by the DSP for decision-making and display.

TRACK-TO-TRACK CORRELATION

ACDS Block 1 is charged with creating a non-redundant track file out of the data produced by all its input sources, which usually arrives in the form of tracks created by external systems rather than raw sensor data. Quantitative performance requirements for correlation have been established. For the case of remote-local (radar-based) tracks, the requirements are expressed as % of dual designations (DD), i.e. redundant data link track reports, and % of false correlations (FC)

for tracked objects spaced at specified statistical distances (functions of sensor standard deviation.) These correspond to Type 1 and Type 2 error limitations. The DD and FC requirements are sufficiently ambitious that they cannot be met simultaneously when correlation/ decorrelation decisions are based on such simple criteria as relationship of position deltas to inner and outer bins. Historically emphasis has been placed on maintaining a low DD rate, since this decreases the traffic burden on the data link, even at the expense of a higher FC rate. Although false correlations often resolve themselves in time as the falsely correlated objects separate in position, an erroneous identification may be produced by mingling the wrong evidence. Thus, in order both to meet quantitative correlation requirements and to increase identification quality, development of a sound technique for track-to-track correlation in a busy environment has been a prime Block 1 concern.

Correlation Algorithms

A common technique for track-to-track correlation was developed which is used for remote/local radar tracks, ESM/radar tracks, and ESM line-of-bearing correlation. In this sequential, multiple hypotheses scheme, candidates for correlation with a new track are selected using a gross correlation test. For each candidate a correlation score is established and revised by filtering as updates are received from the sensors. A correlation is declared when the score rises above a correlation threshold, (and if certain conflict tests are passed, including ID conflict) and is broken when the score falls below a decorrelation threshold. Candidates are eliminated only when the filtered score drops below a purge threshold. Until then the scoring process continues even if another candidate has passed the correlation threshold.

Trade-off studies for choices for threshold values and filter gain, which affect the Type 1 and Type 2 error rates and the time required (number of updates) to reach a decision, were performed. It was determined that Block 1 quantitative requirements for DD and FC could be met through this scheme if the tracks being compared had unbiased position estimates and reasonably accurate covariance data were available. Most of the trackers/systems with which ACDS Block 1 will interface, however, were developed to interface with earlier combat direction systems that did not require accurate track covariances. So, although the systems generally develop and maintain covariances as part of their tracking process, error data is not passed over the interface. This is true even for Link 11 track quality (TQ). Consequently Block 1 contains algorithms for estimating covariances. Also, since relative biases must be expected among tracks produced by distributed systems and hardware, Block 1 includes algorithms for bias estimation and correction. These include estimation of inter-sensor bias, as between the ownship ESM and radar trackers, and gridlock correction.

The automatic ongoing gridlock correction process accounts for remote/local misalignment in x, y, azimuth, and rangescale which are induced by biases in navigation system

estimation of ownship position and radar tracking biases. The process corrects for misalignment with the Link 11 Gridlock Reference Unit and for misalignment with each Link 16 participant (Link 16 has no GRU.) A least mean squares batch processed gridlock correction algorithm is used which requires less computer processing, (CPU) time, with comparable bias reduction performance, than a Kalman filter approach.

Combined DSI/Gridlock Performance

DSI remote/local correlation performance and gridlock performance are interdependent. With large biases remaining between remote and local track reports for the same tracked object, no correlation or false correlations will occur since the decision statistics are taken from an unbiased model of reality. The gridlock pads are estimated using accumulated differences between track data from correlated remote and local reports, thus when incorrect correlation decisions are taken, the gridlock pads will not be accurate.

To evaluate combined DSI correlation and gridlock performance and refine the algorithms, a system model was developed that simulates sensors, data link, and Block 1 track management. A scenario was created to stress both correlation and gridlock, illustrated in Figure 3, that consists of 20 pairs of targets arranged in a spiral and converging on the Gridlock Reference Unit (GRU). Each target pair is represented by a • and the members of each pair are spaced in accordance with a specified performance spacing requirement. Early in the scenario only some of the objects are mutually detectable by the GRU and the other participating unit (PU).

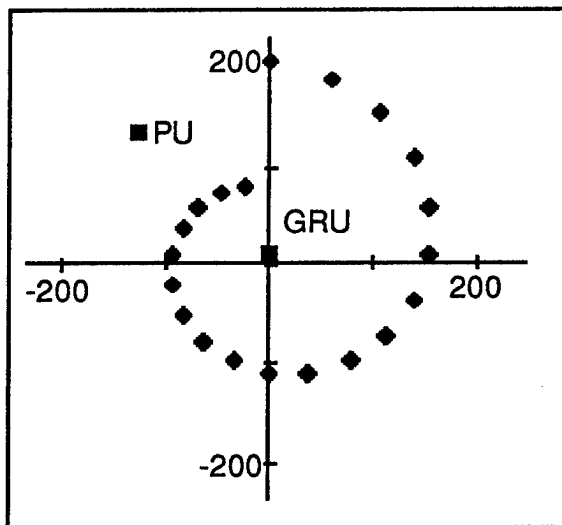


Figure 3: DSI/Gridlock Scenario

Figure 4 illustrates gridlock quality achieved compared with required quality over the scenario life for one model execution. Substantial, but not unrealistic, biases were applied to the GRU's radar azimuth measurements and to the GRU's estimate of its own position in (x,y). The graph illustrates excellent gridlock quality over the scenario life. The quality metric is a normalized measure of the position match of remote/local mutual tracks after gridlock correction pads are applied.

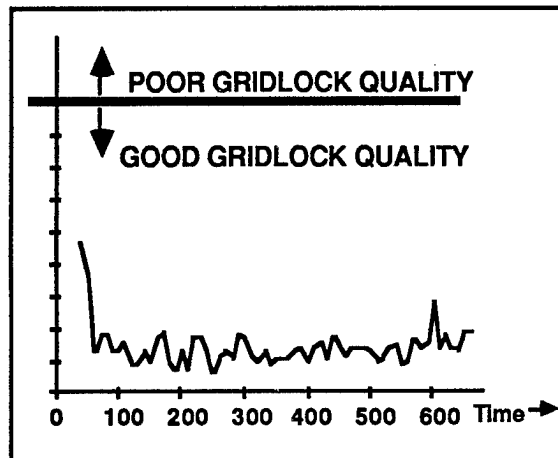


Figure 4: Gridlock Quality

Figure 5 illustrates average (Monte Carlo) correlation performance over the scenario life in terms of % of dual designations (DD), i.e. redundant tracks and % of false correlations (FC). Four scenario cases are shown, identical except for spacing between target pair members. Biases were applied to the GRU as described above. The first case was run using the required spacing for the specified standard case; required performance for this case, in terms of maximum per centages of DD and FC, is shown as a horizontal line. The other cases illustrate performance when spacing between pair members is reduced to three-quarters, one-half, and one-quarter of the required standard spacing. Note that the simulated correlation outperforms the requirement, and is still quite good even when objects are quite closely spaced. This allows considerable margin for at-sea performance with real sensors where a more random mix of spacing is to be expected and clues, such as ID and IFF responses, will be available to aid correlation.

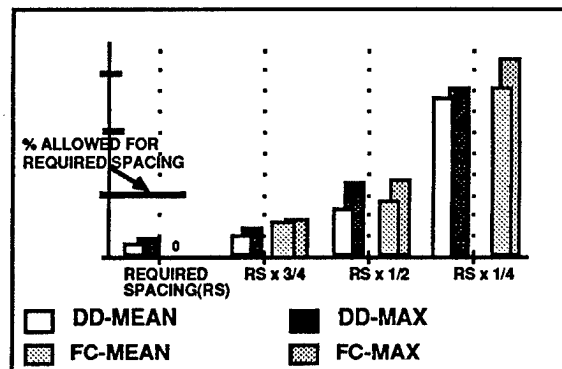


Figure 5: Correlation Performance with Gridlock Bias Correction

ESM Tracking

The ESM SSI distinguishes among: (1) *intercepts*, which are detected emission measurements with an estimated line of bearing (LOB); (2) *emitters*, the physical devices that are emitting; and (3) *platforms*, the ships or aircraft on which the emitters are located. Correlation of two or more intercepts with

natching waveform measurements produces an emitter (x,y) position and velocity estimate and an area of probability (AOP). Spatial correlation of two or more emitters yields an estimated position and classification of the platform carrying them. The quality of the data upon which the ESM SSI operates varies considerably depending on the source. For the most part, however, the bearing measurement data will be imprecise, produced by shipboard or aircraft ESM sensors.

The ESM SSI is required to perform "automated triangulation." It was determined early on, however, that merely finding the intersection point of two bearing lines associated with non-simultaneous detections by different sensors of (presumably) the same emitter was not sufficient to meet the larger goal of classifying non-cooperative radar tracks using associated emission data. It became clear that an ESM tracker was required that would produce unbiased position estimates, good covariance estimates, very good velocity estimates, and recognize and adjust for maneuvers. The velocity estimates are used both for extrapolation and to reduce classification ambiguity when radar-based velocity is not available. The figure below shows simulated tracker performance, using line of bearing reports from two sensors, in following an object through 2 turns executed at 3 degrees per second. Initially, while the tracker settles and the geometry is poor (the bearing lines at first are nearly parallel), the (x,y) positions are imprecise but unbiased, the range estimate is poor, while the bearing estimate is good. Later the tracked position follows the true position closely. Of more importance to correlation, the tracker produces very good estimates of 90% error ellipses and velocity.

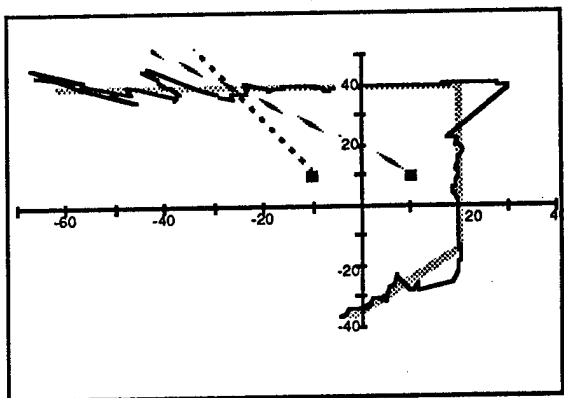


Figure 6: ESM Tracker Performance

MODEL 5 TAXONOMY

Data in the track stores, and corresponding symbology, are organized in terms of a "Model 5" taxonomy defined for use within ACDS Block 1 in order to minimize loss and distortion of data from tactical and intelligence community sources, and maximize compatibility with TADIL J and the latest global data fusion systems such as the Flag Data Display System (FDDS) and the Afloat Correlation System (ACS). Generally, the external systems and libraries with which ACDS Block 1 will interface don't agree on taxonomy structure, parameters, or admissible parameter

values. (Existing tactical identification systems used today by NATO and US tactical forces are defined in the messages used on TADIL A (Link 11) and TADIL B, with JINTACCS and NTISA enforcing interoperability for systems using these data links. The intelligence and over-the-horizon targetting (OTH-T) communities use an entirely different system based on character oriented messages, standardized in the RAINFORM Maritime Reporting System message set.)

The objective of identification processing is to fill in the blanks on a track by supplying a suitable value for each parameter in an identification taxonomy. The ideal taxonomy would be universally useful, providing a complete and unambiguous set of choices for each parameter. The structure would include a hierarchical description capability, ranging from detailed to general where the most detailed level would still be consistent with the kinds of discrimination achievable with the given sensors. The structure would also lend itself to maintaining, independent of a stable description, judgments about temporary track characteristics such as organizational associations and threat potential. And, in fact, the simplified models of reality that are used in algorithmic research make these sorts of assumptions about the taxonomic language behind the algorithms.

Unfortunately, since reality is not neatly structured and our sensors each see reality from a different perspective, none of the taxonomies commonly in use today satisfies all the above criteria. Nor is it likely that a perfect taxonomy could be devised. As an example illustrating the problem, consider the Russian "Bear" aircraft. The name "Bear" calls to mind "Bear bomber", but actually designates both the TU-95 and the TU-142 and, depending on the type modification, indicates either a bomber, reconnaissance or ASW aircraft:

TU-95	Bear A	Bomber
TU-95	Bear D	Reconnaissance
TU-142	Bear F	ASW

The ACDS Block 1, or Model 5, taxonomy structure was developed both to maximize commonality with STANAG 1241 and TADIL J and to overcome well-known limitations in the Link 11 (Model 4) taxonomy. The resulting Model 5 taxonomy provides an integrated track identification that can accommodate data from all sources, including ELINT and rainform-derived classifications. It supports the Composite Warfare Command (CWC) concept by providing information for all categories of tracks. The structure differentiates clearly between reference points and tracks, and describes all tracks in a similar fashion irrespective of category or data source, while minimizing the interaction of fields. This makes consistent MMI for data entry and review possible, no matter what identification data is displayed, which is particularly important for review and creation of doctrine rules for ID, threat-handling, and display filtering.

The taxonomy is organized around three semi-independent axes:

(1) Description answers the question *What is it?* This is an object invariant. The level of detail ranges from category through platform, type, and type modification. Even the

unit -- side number or hull number -- may be discernible by some sensors, such as the human eye, so a place is provided in the taxonomy for maintaining this data.

(2) Threat Potential answers the question *What is it doing?* This includes Identity (Friend, Neutral, Hostile, etc.) and activity and status parameters.

(3) Organization tells to whom the track belongs. It includes Nationality and Alliance parameters as well as the service organizational structure.

Additionally indicators of special considerations, such as exercises, and data sources, such as PPLI (precise position locator message transmitted by Link 16 participants) or IFF, are maintained internally for the purpose of modifying identification or confidence values.

The Model 5 taxonomy structure is based on that of TADIL J (implicit in the message set) which comes close to meeting the desired taxonomy features. Some additional, i.e. not for transmission, parameters have been defined for ownship use only. Admissible values will be a superset of parameter values either defined in the data link message standards or in use by the other systems with which ACDS Block 1 interfaces.

By contrast, the Link 11 standard identification structure used by current "Model 4" tactical data systems such as AEGIS C&D-ADS and NTDS Model 4 is considerably more complicated. Vehicular track data, EW track data, data about link participants, Anti-Submarine Warfare (ASW)-related data, data from intelligence sources, and data about controlled aircraft must each be stored in a different Model 4 format. This data-dependent structure is almost unworkable in a data-fusing system where it is possible, for example, to have one track file record representing an aircraft under the control of an air controller, participating on the data link, responding to IFF interrogation, and detectable by the SLQ-32 ESM sensor. Model 4 track description fields depend on the assigned identity so that a change from *Friend* to *Hostile* can require reformatting the data. The admissible values for Primary Identification (PRI ID) and identification amplification (ID AMP), which in combination determine the base symbology displayed on the console screen, contain a mix of activity and description values; a change in one of these values can require regenerating the displayed symbol for a track. Finally, Model 4 is neither as complete nor as detailed as Model 5 will be. For example, intelligence information is not implemented or integrated with surveillance data, there are no land special points, and much ESM-derived classification data can be transmitted only in the form of emitter numbers. Figure 7 illustrates the increased classification capability possible due to the change in taxonomy from Model 4 to Model 5.

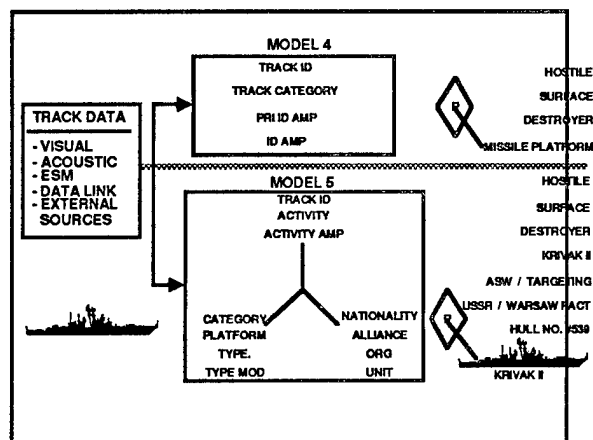


Figure 7: Model 4 and Model 5 Taxonomies

IDENTIFICATION SYSTEM OVERVIEW

The ACDS design shown in Figure 8 provides for a multi-level identification process. The two-tiered integration process has available to it a variety of stored data including: a priori presence probabilities, libraries, speed/altitude relationships, IFF code look-ups, and operator-modifiable ID doctrine rules.

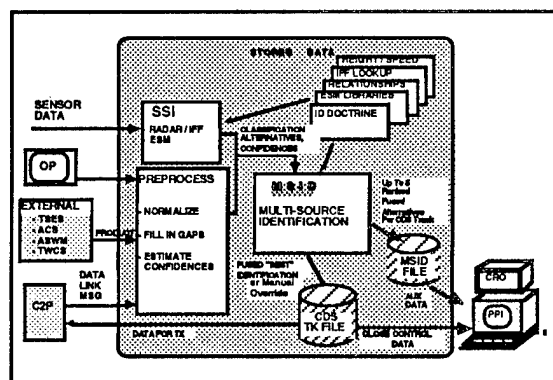


Figure 8: MSID Overview

At the initial fusion level, Similar Source Integration (SSI) functions combine sensor data received from sets of independent sensor data processors with similar characteristics, and derive classification alternatives and confidences.

In the ESM SSI, tracks are classified in one of two ways: (1) either by applying a Bayesian technique together with knowledge about joint distributions of emission parameters (per the ESM libraries) and measurement noise; or (2) by comparison with an Emitter Recognition File which stores electronic signatures of previously seen or anticipated objects. The Emitter Recognition File is useful for storing Friendly Electronic Order of Battle data and for rapid classification of intermittently emitting radars.

In the Radar/IFF SSI, IFF interrogation responses from cooperative objects are interpreted into taxonomy parameter values by accessing an operator-modifiable IFF look-up library. (Reception of codes not in the library

results in prompts to the operator to add to the library.) Confidence is assigned to an IFF-tagged track based on clues such as combination of mode-responses received, match with air plan, compatibility of object kinematics with derived classification, et cetera, i.e. the same sorts of clues customarily used by an operator in assigning identification based on IFF. Classification of non-cooperative tracks to the category level, and sometimes to the platform level, is performed by comparing observed track kinematics with stored speed/altitude profiles.

Also at the initial integration level, other sources, which concentrate on non-real time processing, contribute identification products which must be incorporated into the final identification. These include, for example, acoustic classifications produced by the CV-ASWM. ACDS preprocesses these products so that they also arrive at MSID in the form of alternatives with confidences. The preprocessing includes taxonomy normalization and confidence estimation. Some loss of data and performance degradation occurs which might be avoided if the interfaces were designed to make such processing unnecessary.

At the higher level, after correlation by the Dissimilar Source Integration (DSI) function, the Multi-Source Identification (MSID) function combines classification and confidence data to arrive at fused alternatives and a "best" identification for the track. A Dempster-Shafer approach, selected as most appropriate for combining the classifications thus produced through distributed processing, is augmented with knowledge about known relationships such as weapon-launch platform, basing, and air corridor relationships. The result is a list of multi-level alternatives, with confidences. At this point a ranking algorithm is invoked, which takes into account both confidence and level of detail, to determine which is the best identification for the track. For example, suppose the alternatives for the same track include:

AIR, FIGHTER, MIG 23	Confidence
= 0.45	
AIR, BOMBER	Confidence = 0.49
AIR	Confidence = 1.0

Here the highest confidence alternative -- just AIR -- yields the least information. Which of the other alternatives is best is not at all obvious. The "best" identification thus determined, after consistency and conflict checking, will be entered in the track stores, displayed, and transmitted over the data link.

The ID (Friend, Hostile, . . .) value is assigned by applying data link protocols and operator-modifiable doctrine rules. Because ID is really a political, not a permanent, characteristic of an object, ACDS uses automated doctrine rules for flexibility in adjusting ID assignment to changing rules of engagement.

THE OPERATOR'S ROLE

Ideally an automated identification system would free the operator from dealing with individual tracks, but in reality conflicts and ambiguities will occur. Furthermore, operator data entry will be required for data from

systems lacking a digital interface, such as voice reports. ACDS alerts the operator, as required by data link protocols, to process data link conflicts and provides displays and menus for selection of responses. (Based on the AEGIS experience, such conflicts are to be expected when new, automated systems and the earlier manual systems operate together in the Battle Force.) The authorized operator may also override an automatically supplied identification on any track; the system flags the track so that the automated process will not subsequently override the operator. Similarly, the operator may correlate two tracks; track data comparison displays are provided him. ACDS Block 1 also maintains SSI data which the operator may access to make a decision to decorrelate a source from an already correlated track.

When the operator is acting in his data entry capacity, the identification algorithms require that his input be formatted as alternative(s) with confidence(s). The MMI has been designed with menu choices of taxonomy values and confidence values to foster human-algorithm interoperability, preventing spelling and other legality errors. For example, the operator may cursor to CATEGORY and select *Air, Fighter, Probable* then cursor to TYPE and select *MIG 23, Possible*. If he had already entered US as NATIONALITY, MIG 23 would not appear on his menu while F-4 would.

Controlling Automaticity

The automatic process is driven by algorithms, doctrine statements, and libraries. The operator tunes DSI correlation algorithms by selecting from system settings that correspond to "normal", "slow but sure", and "quick and dirty"; the effect of these controls is to modify thresholds and filter gain. Declaration of track description or nationality results when the identification algorithm-produced confidence exceeds a declaration threshold. The operator tunes the identification algorithms through "certainty controls", selecting the standard, default, setting or "more certainty required" or "less certainty required"; the effect of these controls is to modify the declaration thresholds.

In order to tailor automatic ID-setting to operational policy, the operator may create or modify automatic doctrine statements. For example, "If AIR & USSR & range < 300 NM, then set ID = SUSPECT." Doctrine statements, or automatically fired if-then rules, have been successfully employed by the AEGIS ships; care has been taken to preserve interoperability when implementing them in ACDS Block 1. The "if" clause contains multiple tests of track record contents, such as ID and kinematics, joined by logical connectors "and", "or", and "not". The "then" clause is an action to be performed by the system whenever the "if" test is passed. Track management actions include changing the ID or platform value assigned to a track, and IFF interrogation. (The principal use of automated doctrine in Block 1, however, is for threat ranking and engagement decisions.) ACDS Block 1 provides powerful console-specific display filtering capabilities because its increased surveillance volume and track file capacity contain far more data than can be displayed at any one console. Display filters and alert

criteria (as well as threat evaluation/ weapon assignment rules) are also entered into the system as doctrine statements.

Automated Doctrine

As illustrated in Figure 9 the operator may predefine doctrine statements and store them for activation either singly or as a set. Thus in effect he can reprogram the system to include standard operating procedures.

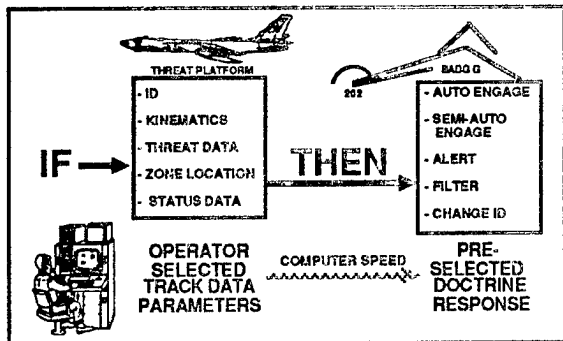


Figure 9: Use of Automated Doctrine

SOFTWARE TECHNIQUES

The ACDS Block 1 program will be manipulating vast quantities of data to meet real-time tactical needs. In order to handle the associated requirements for computation, rule-firing, search, and display, a new way of organizing the software program was required. Block 1 features many techniques which, though proven in commercial applications, are innovative for Navy software programs. Figure 10 illustrates the approach.

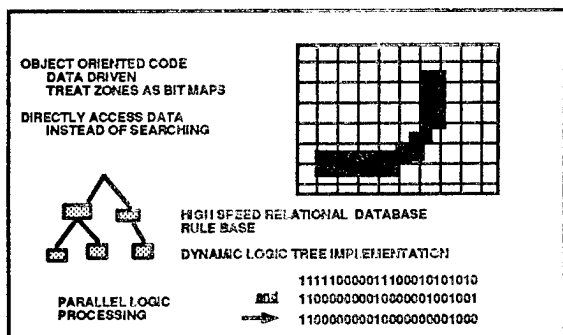


Figure 10: Block 1 Software Features

The program is organized as data-driven, object-oriented code and contains an Object Management System (OMS). Whenever possible "adaptable parameters" (input data) are used instead of hard-wired data. This reduces the numbers of lines of code that need to be written and tested and makes it easier to adapt the program to future needs. For example, display formats are defined in terms of track store objects and adaptable parameters and thus can easily be reformatted. Large portions of the libraries which are stored on disk are also represented in active memory as objects with pointers to their disk location; this speeds up the process of reading from disk and eliminates many reads altogether. The "if" conditions for

the automated doctrine rules may reference zones; for example, "if an aircraft exits a commercial air corridor (a zone), then . . . ". The zones also are defined as bit-map objects, rather than in terms of equations of lines and circles; location of a track object inside or outside the zone can then easily be determined by set logic operations rather instead of by expensive computation. Track stores and libraries are constructed as relational databases and the program features a high-speed query process. The doctrine rule base contains both active and inactive rules; active rules are combined via a dynamic logic tree.

ISSUES

The ACDS algorithms are expected to properly associate tracks and then produce valid identification results speedily to support real-time tactical decisions. But decision quality and the quantity of tracks that can be correctly correlated and identified are affected adversely by problems and issues beyond the internal ACDS Block 1 design, notably:

- (1) Interfacing systems and libraries each speaking a different language, where no language is complete or perfectly structured, resulting in loss of data, ambiguity, and extra processing;
- (2) Systems trading identification products resulting in the loss of statistical independence;
- (3) Model 4 manual systems (or other automated systems using different algorithms, including PC programs), operating in conjunction with Model 5 automatic systems, producing conflicting identification or correlation results;
- (4) Costly workarounds and estimation procedures when good data and error estimates maintained internally by subsystems are not provided across interfaces to the data fusor.

The proofs or justifications for algorithms that combine classification and identification data from multiple sources are based on assumptions about the data which is to be combined. Ideally the data provided to the ACDS Block 1 MSID process would fit those assumptions. That is, the data: (1) would have been derived independently by the several sources; (2) would be equally valid for each source; (3) would be reported in a common language; (4) would include all plausible classification alternatives; and (5) would assign confidences that have the same meaning to those alternatives. (For example, if one system uses zero confidence to denote certain impossibility, another system should not use zero confidence to denote lack of knowledge.)

The reality of the situation faced by MSID is far from ideal. As indicated in Figure 11, many of the systems which provide data and products to ACDS also trade data and products with each other. ACDS can neither know nor control whether the data it processes from an external source represents new, independent evidence. Furthermore, the existence of identification fields; or even identification confidence fields, in an IDS message is no

guarantee that the content of the message represents identification/confidence data of the sort ACDS expects to process. As a result, ACDS must cope by preprocessing the data not only to normalize the taxonomy values as previously discussed, but also to assign or scale confidences and attempt to fill in the blanks.

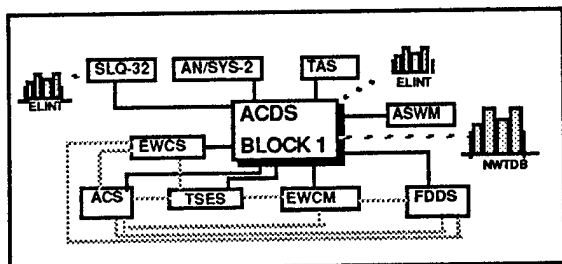


Figure 11: Interfacing Systems

The other systems in the web, many of which are still in early development, will also have to deal with this problem. SPAWARS has been reexamining shipboard system architecture in order to consolidate and eliminate overlapping functions. NATO working groups continue efforts at standardization, not only of the taxonomy, but also of the format for interchange of statistical confidence related to identification information. In the long-term the solution must be to address data integrity as an issue and include safeguarding requirements when new systems and architectures are devised.

CONCLUSION

ACDS Block 1 was designed to meet the surface Navy's need for data fusion in the present and future operating environment. The program is responsive to fleet requirements for increased range and track capacity and improved integration and automation. It produces a comprehensive tactical picture using high performance algorithms and assists the operator in evaluating that picture through operator specified automated doctrine. But ACDS Block 1 is only one element of the total shipboard and battle force system. Its performance depends on the quality of its input systems and on the architecture that links it with other systems that are also performing data fusion. It remains to be seen whether the combat system architecture of the future will promote good fusion performance or adversely affect it.

As Admiral C.A.H. Trost, Chief of Naval Operations, said, "It is for consideration whether we will have the discipline and the foresight to build one fused system, with no matter how many peripheral components, or have the equivalent of a couple of hundred individual personal computers that cannot talk to each other."

The Advanced Combat Direction System Block 1 is the first major step in building that fused system.

THIS PAGE LEFT BLANK INTENTIONALLY

DYNAMIC COORDINATE-TO-FEATURE ASSOCIATION
FOR COMMAND AND CONTROL APPLICATIONS

Terence M. Cronin

US Army CECOM Center for Signals Warfare
Vint Hill Farms Station
Warrenton VA 22186
Phone: (703)-347-6430

Abstract: There is an alarming lack of technology available for automated systems to perform the sophisticated high level reasoning necessary to react quickly to reports pertaining to the location and posture of opposing forces. At the most fundamental level, it is essential to be able to rapidly process an arbitrary map coordinate by associating it with the spatial feature(s) nearest at hand, especially when feature boundaries vary in time. The US Army Center for Signals Warfare has conceptualized a theory of front end spatial perception which provides a solution to this problem. The theory has produced three new computer algorithms called *loci-reduced spatial discrimination*, *annulus-based inclusion testing for multiply-connected sets*, and *topographical contour betweenness testing*. The first two techniques exploit data structures called the *equidistance loci set* and the *inner annulus* as respective criteria for nearness and closed-contour inclusion. Building upon loci reduction, the last technique extends the first technique into the third dimension to reason about a coordinate's directional gradient. On maps of real world boundary complexity, the theory has facilitated the automation of the following spatial reasoning computations in real-time: nearest (time-varying) feature to an arbitrary coordinate; distance of a feature from an arbitrary coordinate; relative direction of a feature from an arbitrary coordinate; a decision about whether an arbitrary coordinate is inside or outside a feature characterized by a multiply-connected closed contour; the two elevation contours between which an arbitrary coordinate lies; the local directional gradient of an arbitrary coordinate. When combined in a package, loci reduction, annulus-based inclusion testing, and topographical contour betweenness testing provide a powerful toolset of utility functions for command and control map reasoning processes.

DYNAMIC COORDINATE-TO-FEATURE ASSOCIATION FOR COMMAND AND CONTROL APPLICATIONS

Terence M. Cronin

US Army CECOM Center for Signals Warfare
Vint Hill Farms Station
Warrenton VA 22186
Phone: (703)-347-6430

INTRODUCTION

The process of reasoning about the location and posture of opposing forces is of paramount importance to the success of friendly force command and control operations. Actions taken by either side are a function of how force elements are positioned with respect to both natural and manmade spatial features. At the perceptual level, an automated battlefield map reasoning tool must be capable of spatially associating a coordinate with the geographic features and force elements nearest at hand. The process should also be able to perform inclusion testing on features which are contained within other features, and should be able to understand the implications of local topography upon communications and maneuvers. Such a low level perceptual capability is required across a wide range of battlefield management applications, including intelligence production, situation assessment, and command and control. It is futile to attempt higher level reasoning for such applications without possessing the faculty to make snap judgments at the spatial perception level.

THREE NEW SPATIAL PERCEPTION ALGORITHMS

This paper describes three new algorithms designed to solve low-level spatial reasoning problems on a map. Each algorithm is designed to emulate a specific map reasoning function of the kind that a human can perform at a glance. The first algorithm, called *equidistance loci-reduced spatial discrimination*, is designed to associate an arbitrary coordinate with the nearest spatial feature, under conditions when feature boundaries are time-varying. The second algorithm, called *annulus-based inclusion testing for multiply-connected sets*, is designed to decide if an arbitrary coordinate is inside a closed contour (and if so, where inside), when the contour may be multiply-connected. The third algorithm, called

topographical contour betweenness testing, is designed to label an arbitrary coordinate with the two topographical elevation contours between which it is situated, and also to render the local gradient based on interpolation.

A tabulation of the techniques is presented below at Table 1. This treatise is kept intentionally at a tutorial level: the mathematical development of the algorithms is presented elsewhere (Ref. 2-4). It should be emphasized that each technique is deterministic (i.e., computes a single, certain solution). The three problem areas succumbed to deterministic solutions because a geometric approach produced compact data structures which partition a map based on the spatial orientation of its features. The discriminating boundary of the partition is small in size when compared to the entire map, and the resultant data compression guarantees feasible polynomial complexity; indeed, an implementation has demonstrated that each algorithm runs in real-time.

Table 1. Three New Algorithms for Command and Control Reasoning.

Cartographic Problem Area	Algorithmic Solution
Find nearest time-varying features to an arbitrary Coordinate Return as a Vector the Distance and Direction to the features	<i>Equidistance Loci-Reduced Spatial Discrimination</i>
Decide if a coordinate is inside a closed curve which contains other closed curves	<i>Annulus-based Inclusion Testing for Multiply-connected Sets</i>
Determine between Which Two Elevation Contours a Coordinate Lies Decide the Local Slope and Orientation of the Coordinate	<i>Topographical Contour "Betweenness Testing"</i>

CLOSEST-POINT PROBLEMS

Finding the closest point to a given point is a

well-studied problem in computer science. The literature documents several approaches, including Voronoi diagrams (Ref. 12,15) and chamfering (Ref. 1,9). However, none of the techniques directly addresses the problem of locating the nearest *time-varying feature boundaries* to an arbitrary coordinate. The concept of distance is well-developed; for an informative treatment see reference 10. The distance which will be used here is variously known as the *city-block* distance, the *Manhattan* distance, or the d_4 distance (Ref. 13). The d_4 distance between two points $P=(x_1,y_1)$ and $Q=(x_2,y_2)$ is defined as follows:

$$d_4(P,Q) = |x_1 - x_2| + |y_1 - y_2| \quad (1)$$

If a Euclidean metric is absolutely required, the d_4 distance can first be used as a filter to find the closest point to a coordinate; subsequently only the Euclidean distance to that point alone need be computed. In this fashion, relatively expensive floating point operations of exponentiation and multiplication are minimized.

The Loci-Reduced Spatial Discrimination Algorithm

Equidistance loci reduction discriminates nearness to a feature by fencing in the feature with the equidistance loci contour induced by considering the feature pairwise with every other feature on the map. The concept is informally described as follows. Select an arbitrary point on the boundary of some feature. Consider moving orthogonally away from the tangent of the point with the contour until the halfway point to the nearest feature (or the edge of the map) is encountered. Remember the halfway point. Continue the process for every point of the feature boundary. The resultant set is called the *equidistance loci contour* for the feature. Now repeat the operation for every other feature in the map space. Unite the equidistance loci contours across all features - the union is called the *equidistance loci set* for the map feature space. An example of a loci set is shown at Figure 1. In practice, the outline of a feature is traced in over a digitizing map with a man-machine interface. A smoothing algorithm removes gaps created by a user's hand moving faster than the video sampling rate. The resultant linked list of contiguous coordinates is stored internally by the computer, and is called the *trace contour* associated with the feature.

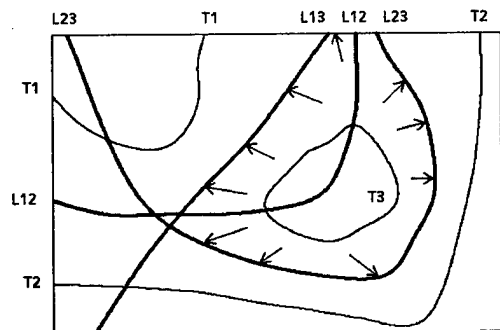


Figure 1. The loci set (in bold) displayed for a map space containing three features. The loci contour for feature T3 is delineated by the line encompassing the arrow tips. Any coordinate within this contour is nearer feature T3

than any other feature.

The Computational Complexity of Loci Reduction

Theorem. Equidistance loci reduction is of feasible polynomial complexity.

Proof: Assume a binary map space of n features, each represented by a trace contour of contiguous Cartesian coordinates. From an arbitrary coordinate (x,y) the algorithm must perform $\sum |T_i|$, $i=1, \dots, n$ distance calculations of the form: $|x - x_j| + |y - y_j|$, and also for each feature T_i must perform $|T_i|$ comparisons to locate the smallest such distance. In addition, n comparisons are required to locate the smallest distance across n features. Since the expressions do not contain n as an exponent, the algorithm is polynomially complex. Furthermore, the algorithm uses integer-valued coordinates; packs both x and y coordinates into a single computer word; and avoids operations higher in the computational hierarchy than addition, subtraction, and absolute value. Therefore, loci reduction is of *feasible* polynomial complexity.

A Geometric Nearness Criterion

The equidistance loci set defined on a feature space provides a geometric rationale for employing a nearest-neighbor technique. Nearness of an arbitrary coordinate to a feature is defined naturally by minimizing the distance across all trace contours. This process is conceptually equivalent to checking to see if a point lies within the bounds of the equidistance loci contour of a feature.

Definition. A point P is *nearest* to feature F_i with associated trace contour T_i iff $d_4(P,T_i) < d_4(P,T_j) \forall j \neq i$.

Definition. An object *moves* from the vicinity of feature F_i to the vicinity of F_j iff the object crosses loci contour L_{ij} .

Contrasting Loci Reduction with Voronoi Diagrams

The equidistance loci reduction process induced by a feature boundary space is distinct from the *Voronoi diagram* induced by a point set for two related reasons. First, loci reduction operates on sets of *curved, contiguous boundary* point sets, whereas the traditional Voronoi diagram constructor operates on random point sets and simple polygons. Secondly, loci reduction does not look for *all* points closer to a given point than any other point; indeed, it may be argued that the contiguous neighbors on a connected boundary meet this criterion. Rather, given an arbitrary map coordinate, loci reduction searches every feature boundary for the point closest to the coordinate, and then selects *the* point which is closer than all the others. The process is repeated for every point on the feature boundary. It is clear that the feature boundary space processed by equidistance loci reduction is more general than the random point sets or simple polygons processed by traditional Voronoi diagram constructors.

Contrasting Loci Reduction with Chamfering

Chamfering is a technique which computes nearness by exploiting the speed of prearchived table lookup. On a map containing $m \times n$ pixels, for each feature an array of $m \times n$ cells is precomputed prior to run time, where each cell contains the distance to the feature. The technique is efficient for manageable numbers of static features, since it avails itself of table lookup, which is a random access operation. Regrettably, in scaled-up domains, the number of spatial features rapidly becomes unwieldy. For example, a map space of size 700X700 pixels with twenty binary features uses twenty arrays, each containing 490,000 bits. As the number of features increases, paging becomes inevitable, which leads to a thrashing problem when chamfer arrays must be continuously swapped back and forth between physical and virtual space. Another drawback of chamfering is that when a feature boundary alters in shape or position, a new corresponding chamfer array must be computed, which is a time-consuming operation unsuitable for many applications.

CLOSED CONTOUR INCLUSION PROBLEMS

The Odd/Even Crossings Technique

Deciding whether a point is inside a closed polygon is another problem area which has been studied, but not to the degree required to treat the multiply-connected sets which frequent maps of real-world complexity. There is an algorithm available which counts the number of even or odd crossings when a line is drawn from the point through a polygon; if even, the point is outside - otherwise it is inside (Ref. 12,15). The algorithm is an elegant solution for simply-connected sets, but fails for multiply-connected sets. This is a problem when working with maps, since maps are notoriously multiply-connected. As an example of the odd-even crossing technique failing in practice, consider the case of the Great Salt Lake in Utah. Points interior to the lake would generate an even number of Utah crossings, resulting in a decision that points on the Great Salt Lake are exterior to Utah. Although this decision is correct from a topological stance, it is flawed pragmatically. Another limitation of the algorithm is that it relates only whether a point is inside or not - it does not provide any information about *where* inside.

The Annulus-based Inclusion Testing Algorithm

The inner annulus of a closed contour is conceptually simple to describe. The annulus is a data structure generated by traversing the inside edge of a closed clontour in a counterclockwise fashion, and collecting the points visited during the traversal. The key idea is that when one moves along in a counterclockwise fashion, the inside of the contour is always to the left. Once the annulus is generated, it is a simple matter to compare the distance from an arbitrary point to the contour with that to the annulus; if the latter is smaller, the point is decided to be inside (Figure 2). As part of an affirmative inclusion decision, the technique also provides both the distance and direction to the nearest point on the inside edge of the closed contour.

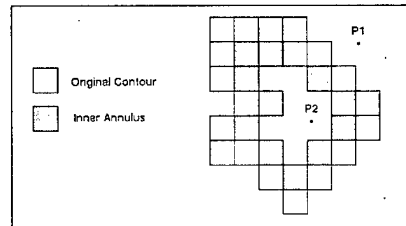


Figure 2. A Graphic Illustrating the Utility of the Inner Annulus. P2 is nearer the inner annulus and therefore inside; conversely, P1 is outside.

The Formal Design Specification of the Annulus

Inner annulus generation is algorithmic. Refer to reference 3 to review the set of mathematical equations specifying the software design of the algorithm. The technique is described here informally. Given a closed digital contour, the first step is to order the pixels in a counterclockwise direction. Starting at an arbitrary point, move along the contour and assign to the annulus the pixel 4-connected to the left of the current pixel. Continue until the start point is revisited. The set of all points produced during the traversal is called the set of *first order generator outputs*.

There are special cases which require second order treatment, due to peculiar morphological structures which give rise to contour shape. First, a *convex corner triplet* is an outside corner of the contour; i.e., it is a set of three pixels such that the middle pixel forms a corner with its 4-connected neighbors when traversed in a counterclockwise direction. Convex corner triplets are problematic because they produce spurious points located on the original contour. Let T^* represent the set of all *first* points of convex corner triplets. Next, a *local concavity* can be conceptualized as a "dent" in the original contour. Such phenomena require second order operators to add additional points to the first order annulus outputs. Let C denote the set of points added. Finally, the annulus generator function may produce pixels multiple times - these points should be entered only once into the annulus structure. Let $D(k)$ denote the set of pixels produced at least k times, where k is between 2 and 4 inclusive.

A Conjecture: The Length of the Inner Annulus

For storage requirement considerations, it is of interest to derive the length of the annulus as a function of the length and shape of the original closed contour. Suppose that the original closed contour is of length n pixels; contains $|C|$ local concavities; $|T^*|$ convex corner triplets; and that the number of pixels produced at least k times by the annulus generator function I_g is equal to $|D(k)|$. It is conjectured from empirical evidence that the length of the inner annulus is equal to the following expression:

$$|I_g| = n + |C| - |T^*| - \sum_{k=2}^4 |D(k)| \quad (2)$$

An example of inner annulus generation and respective validation of the conjecture is shown at figure 3.

It is further conjectured that the length of an outer annulus O_g (a structure for which the contour itself is the inner annulus) is an expression of the same form, with the arithmetic signs of the concavity and convexity expressions reversed:

$$|O_g| = n - |C| + |T| - \sum_{k=2}^4 |D(k)| \quad (3)$$

Informally, the rationale for this conjecture is that what is a concavity from the inside of a contour is a convexity from the outside.

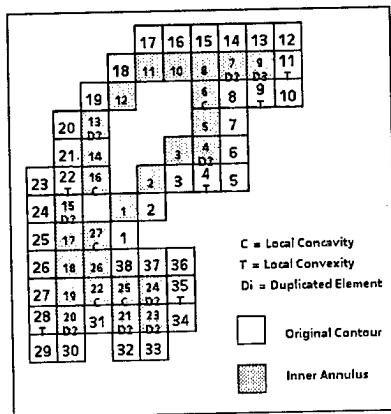


Figure 3. An example showing the generation of the inner annulus for a closed contour of length 38. There are six convex corner triplets, five local concavities, nine points visited at least twice, and one visited at least three times. Conjecture 2 predicts the length of the annulus to be $38 - 6 + 5 - 9 - 1 - 0 = 27$, which is in fact the result obtained in practice.

TOPOGRAPHICAL BETWEENNESS

Computing the local orientation of an arbitrary point on a topographical map is a problem area which has not received the attention paid to either closest-point or closed contour inclusion problems. The topographical orientation problem may be posed as follows: given a topographical map with an elevation contour resolution of k units, return the elevations of the two contours between which an arbitrary point lies; also return an interpolated local gradient, and the direction to the higher of the two contours.

Currently available algorithms which perform elevation reasoning are implemented essentially in the same manner as chamfering is for closest-point problems: the elevation values at each pixel in a digital map are prearchived; table lookup is used at run-time to render the elevation at a particular point. There is a need for higher order logic to reason across elevation contours.

The Topographical Contour Betweenness Algorithm

Topographical elevation contours are in fact spatial features, similar to political or geographic boundaries. As such, they can be traced over a map, and therefore will succumb to the contour entry process described above in the section which introduces the loci reduction process. The "betweenness" algorithm actually makes two calls to the loci reduction algorithm to return the nearest two elevation contours to a coordinate.

Regrettably, nearness is not sufficient to guarantee betweenness. However, the nearer of the two contours is guaranteed to bound the coordinate's elevation from either above or below (for a proof see reference 4). Which way the coordinate is bound is a function of whether or not the line segment drawn from the coordinate to the second contour crosses the first (nearer) contour. If the segment does not cross the first contour, then the coordinate is between the first and the second contour. If the contours are of equal elevation, then the coordinate is on a saddle or in a culvert. If the line segment crosses, and the first contour is higher than the second, then the coordinate is between the first contour and one which is k units higher (if a higher one does in fact exist - otherwise the coordinate is on high ground); if it crosses and the first contour is lower than the second, then the coordinate is between the first contour and the one which is k units lower (if a lower one does in fact exist - otherwise the coordinate is on low ground). Figure 4 depicts the logic flow of the topographical contour betweenness algorithm.

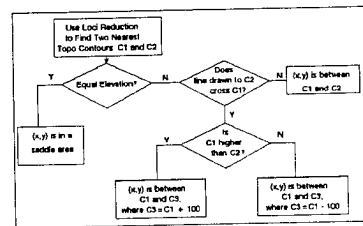


Figure 4. A Flowchart of the Topographical Contour Betweenness Algorithm.

INTEGRATING THE ALGORITHMS

Map reasoning requires an understanding of feature orientation and nearness, an appreciation of how one spatial feature can be part of another, and a capability to decipher a maze of topographical contours into the corresponding three-dimensional reality. Equidistance loci reduction is a geometric interpretation of feature orientation and nearness. Annulus-based inclusion testing for multiply-connected sets is a topological tool which treats the set inclusion problem. Betweenness testing is a geometric tool which addresses the point-to-point directional gradient problem.

When packaged together, the three algorithms comprise a set of tools which can *deterministically* answer many of the low-level spatial reasoning questions which are taken for granted by human analysts. For a command and control operation, which has a requirement to know how both red and blue forces

are oriented upon a map of spatial features, the toolkit provides a real-time capability to put a universal transverse Mercator projection (UTM) coordinate into context, based on the distance and direction to objects in the vicinity. Because the algorithms are of generic mathematical design, they are immediately leverageable against any digitized map of interest.

IMPLEMENTING THE PACKAGE

The three algorithms have been implemented within an Interlisp environment, and have been applied to two styles of binary maps. The first set of maps, used to demonstrate loci reduction and annulus-based inclusion testing, is from the World Data Bank II (WDBII), which contains features consisting of political borders, coastlines, rivers, lakes, and mountain chains. The second set of maps, used to demonstrate the topographical contour betweenness algorithm, is from the United States Geographical Service (USGS) archives, and in addition to portraying a higher resolution version of the same features contained in WDBII, the set includes buildings, roads, and 20 foot elevation contours.

The maps employed are of real world complexity; i.e., a single map displayed on half a CRT screen consumes 500,000 bits of information, and depending upon the local geographic and political profile, the number of features can be large. Although only three WDBII maps and one USGS map have been analyzed, the performance of the algorithms is timely. The response to a typical query requires five milliseconds of CPU time. The performance potential of the map reasoning package has attracted the attention of operations personnel, and as a result the developmental software for loci reduction and annulus-based inclusion testing has recently been ported to advanced development.

CONCLUSIONS

Three new algorithms have been developed to emulate specific functions of the spatial perception process experienced by humans when interpreting a map during command and control processing. The first algorithm, called *equidistance loci reduction*, partitions a map with a discriminating boundary consisting of the equidistance locus formed by considering each feature pairwise with every other feature on the map. This technique is designed to associate an arbitrary coordinate with the nearest spatial feature, even as feature boundaries vary in time. The second algorithm, called *annulus-based inclusion testing for multiply-connected sets*, provides a technique to decide if a point is inside a closed contour, when the closed contour itself contains other closed contours. The technique is important to command and control because maps are multiply-connected. The third algorithm, called *topographical contour betweenness testing*, is designed to decide between which two elevation contours a point sits, and as a corollary computes the local directional gradient. Taken as a package, the set of algorithms deterministically answers many of the

perceptual level questions which a command and control system requires to progress with higher level reasoning. Since the algorithms are of feasible polynomial complexity, the compiled implementation returns spatial snap judgments in real-time, which is a requirement for front-end perception during command and control processing.

ACKNOWLEDGEMENTS

The research benefited from technical discussions with Gerald Andersen, Richard Antony, Christopher Bogart, Thomas Garvey, Henry Kyburg, James Mulligan, Azriel Rosenfeld, and Robert Sedgewick.

BIBLIOGRAPHY

1. H.G. Barrow et. al., "Parametric Correspondence and Chamfer Matching: Two New Techniques for Image Matching", Proceedings of the Fifth International Joint Conference on Artificial Intelligence, Cambridge MA (1977).
2. Terence M. Cronin, "Loci-reduced Spatial Discrimination", US Army Center for Signals Warfare Technical Report, 1988.
3. Terence M. Cronin, "Annulus-based Inclusion Testing for Multiply-connected Sets", US Army Center for Signals Warfare Technical Report, 1988.
4. Terence M. Cronin, "Topographical Contour Betweenness Testing", US Army Center for Signals Warfare Technical Report, 1988.
5. Martin A. Fischler, "Interactive Aids for Cartography and Photo Interpretation", SRI International Technical Report, October 1979.
6. James. D. Foley and Andries Van Dam, Fundamentals of Interactive Computer Graphics, Addison-Wesley, Reading MA (1983).
7. Thomas D. Garvey, "Evidential Reasoning for Geographic Evaluation for Helicopter Route Planning", IEEE Transactions on Geoscience and Remote Sensing, May 1987.
8. Thomas D. Garvey, "Perceptual Strategies for Purposive Vision", SRI International Technical Note 117, September 1976.
9. Thomas D. Garvey and Martin A. Fischler, "Machine-Intelligence-Based Multisensor ESM System", SRI International Technical Report AFAL-TR-79-1162, March 1979.
10. Rafael C. Gonzalez and Paul Wintz, Digital Image Processing, Second Edition, Addison-Wesley, Reading MA (1987).

11. Petros A. Maragos and Ronald W. Schafer, "Morphological Skeleton Representation and Coding of Binary Images", IEEE Transactions on Acoustics, Speech, and Signal Processing, VOL. ASSP-34, No. 5, October 1986.

12. Franco P. Preparata and Michael Ian Shamos, Computational Geometry, Springer-Verlag, New York NY (1985).

13. Azriel Rosenfeld and A.C. Kak, Digital Picture Processing, Vol. II, Second Edition, Academic Press, NY (1982).

14. Eric Saund, "Abstraction and Representation of Continuous Variables in Connectionist Networks", Proceedings of the Fifth National Conference on Artificial Intelligence, Philadelphia PA (1986).

15. Robert Sedgewick, Algorithms, Addison-Wesley, Reading MA (1983).

16. Isaac Weiss, "3-D Shape Representation by Contours", Proceedings of the Ninth International Joint Conference on Artificial Intelligence, Los Angeles CA (1985).

THIS PAGE LEFT BLANK INTENTIONALLY

DECISION AIDING: AN INTERACTIVE TRACKING AND IDENTIFICATION SYSTEM

Dr. Ivan Kadar and Dr. Manikant Lodaya

Grumman Aircraft Systems
Bethpage, New York 11714-3582

ABSTRACT

A new approach to resolving ambiguities of a numerical tracker provides enhanced tracking capability and yields reduced operator workload. This approach utilizes combined symbolic and numerical algorithms which interact with an identification system based on Dempster-Shafer theory. Simulation results demonstrate successful performance of the system.

DECISION AIDING: AN INTERACTIVE TRACKING AND IDENTIFICATION SYSTEM

Dr. Ivan Kadar and Dr. Manikant Lodaya

Grumman Aircraft Systems
Bethpage, New York 11714-3582

INTRODUCTION

This abridged paper illustrates the performance gains achievable by integrating expert systems and numerical tracking algorithms. The resultant tracking expert system (ES) reasons with heuristic rules based on target identification (ID), threat level, collateral information, kinematic ID discriminants, and operator's knowledge.

Target ID information is derived from sensor reports as well as from collateral information, operator heuristics, and target group (kinematic) behavior. This aggregate data is processed by a separate ID-ES based on Dempster/Shافر (D/S) theory.

In addition, this paper illustrates that the combined system can reduce spurious tracks not resolvable by the numerical tracker alone. It should be noted that this system concept is a forerunner to an "intelligent tracker" which will utilize temporal and spatial reasoning along with contextual knowledge to resolve track ambiguities.

BACKGROUND

Most numerical tracking algorithms (Ref. 1) are unable to resolve closely spaced target tracks, such as crossing targets in unfavorable sensor coordinates or during evasive target maneuvers. In these circumstances, one target may be associated with multiple tracks, one track may be associated with multiple targets, or a target may be incorrectly assigned to a track.

Yannone (Ref. 2) discusses recent trends in multisensor/multitarget tracking using probabilistic association combined with artificial intelligence (AI) techniques. Bonissone (Ref. 3) developed a tracking ES to resolve track ambiguities but did not use ID information. Kadar (Ref. 4) developed an iterative evidential reasoning scheme based on a perceptual reasoning paradigm to enhance the performance of both Situation Threat Assessment Response Strategy Systems (STARS) and the ID and tracking subsystems thereof.

This paper generalizes previous approaches by the introduction of an interactive tracking ES and ID-ES.

APPROACH

The decision aid scheme introduced in this paper is different from previous approaches. It consists of interactive tracking ES and ID-ES that reduce both track and ID ambiguities by the combination of numerical and symbolic algorithms. By using kinematic ID discriminant data to amend the input vector of the ID system (concatenating the ID sensor data input vector with the kinematic discriminants), the performance of the iterative system (Ref. 4) can be enhanced with reduced iteration steps. Track kinematic ID discriminants are derived from expected measurement parameters such as target speed and altitude regions, maneuvers, and target location with respect to an apriori-known flight corridor, etc. Thus the inputs to the ID-ES consist of probabilistic declarations based on comparing the extracted sensor data feature parameters with an apriori correlation data base.

The rule-based tracking ES is developed to assist numerical algorithms (branch and bound and nearest neighbor), implemented at the Grumman Artificial Intelligence Lab, to reduce track ambiguities. These heuristic rules are based on experienced-pilot knowledge, target ID and threat priority information, common sense, and collateral information. The tracking ES interacts with the numerical tracking routines and the identification ES. Figure 1 shows the conceptual system diagram.

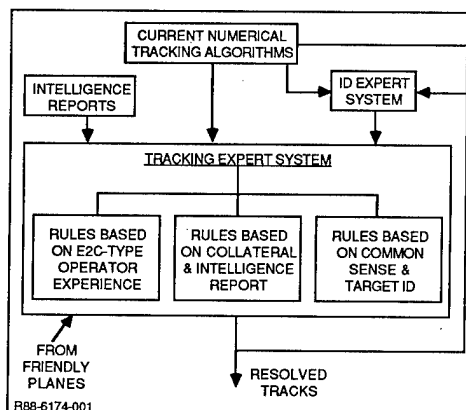


Fig. 1 Tracking Expert System

The ID-ES block diagram is depicted in Fig. 2. This system provides ID of targets by combining information from multiple sensors. The target-feature-discriminant parameters are extracted from individual sensor declarations, including the kinematic features described before, and are compared with a sensor/target feature parameter correlation data base (intelligence files). The correlation process is uncertain due to both measurement noise and the non-uniqueness property of the correlation data base.

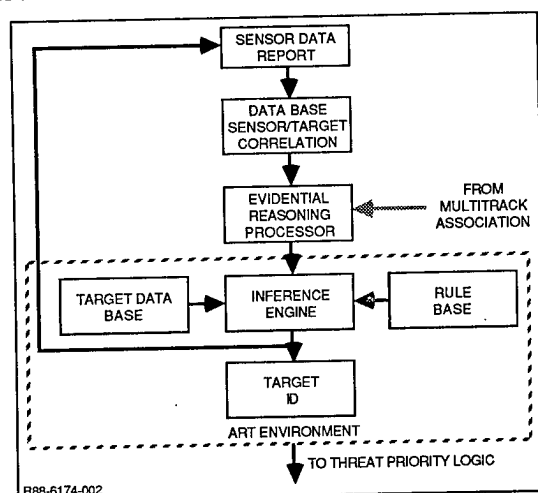


Fig. 2 Target Identification Expert System

The system simulates extracted target-feature parameters that can be derived from sensor measurements/reports. The system assigns a distribution functional to each target-feature parameter. Targets to be identified have their respective feature-discriminant parameters stored in the target data base.

The uncertainties among the target(s) and associated feature(s) are computed based on sensor uncertainties and simulated feature values.

The D/S evidential reasoning scheme is used to combine the uncertain evidencies based on emulated sensor reports. This data is used in the correlation process to yield a set of possible beliefs assigned to targets.

Both iterative (Ref. 4) and noniterative evidence combination schemes are implemented. ART is used in this case to implement heuristic ID rules and to store the target knowledge base (KB). The D/S algorithm is implemented in LISP. The kinematic feature-discriminant data from the tracking ES is used to concatenate the feature vector data based on ID declarations in the absence of track data, as described previously. Thus the ID-ES and the tracking ES interact in a manner to reduce ambiguities both in track and ID.

PERFORMANCE EVALUATION

Figure 3 displays simulated outputs from branch and bound and the nearest neighbor tracking algorithms. The nearest neighbor tracks continue in a straight line after tracks cross (track 3 and track 4). However, tracks generated from the branch and bound routine change direction after they cross (track 1 and track 2).

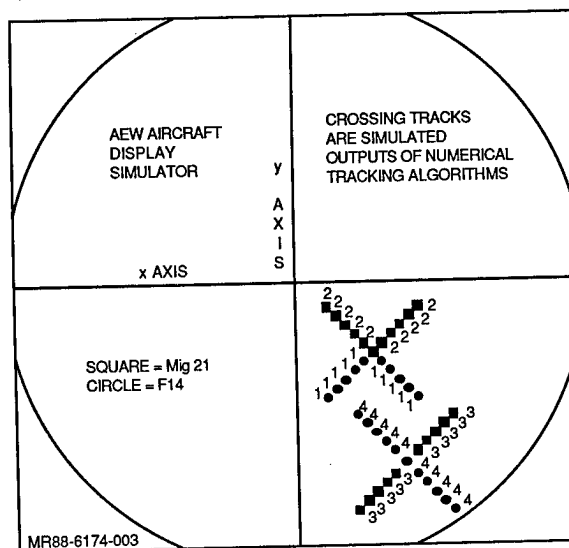


Fig. 3 Crossing Tracks before Expert System Is Run

Figure 4 displays tracks after the tracking expert was run. In this case, track ambiguities based on target ID were resolved. Track 1 and track 2 actually do not change direction, but fly in a straight line after they crossed, based on their ID. This information is passed back to the numerical tracker and ID expert system.

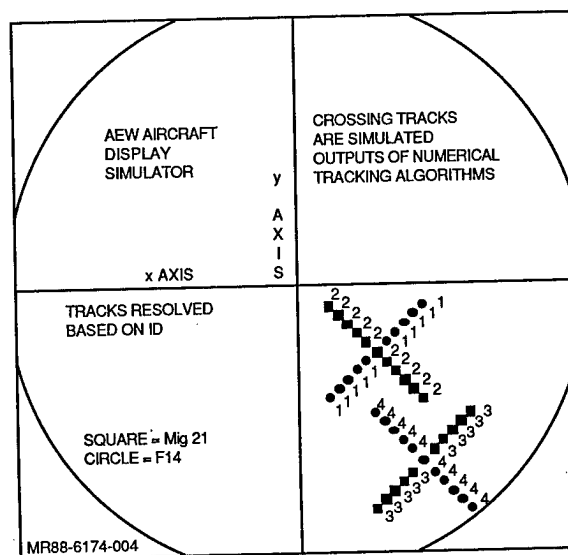


Fig. 4 Crossing Tracks after Expert System Is Run

The ID-ES output is shown in Fig. 5. This result is based on simulated inputs from six probabilistic declarations derived from a sensor suite (both active and passive sensors)

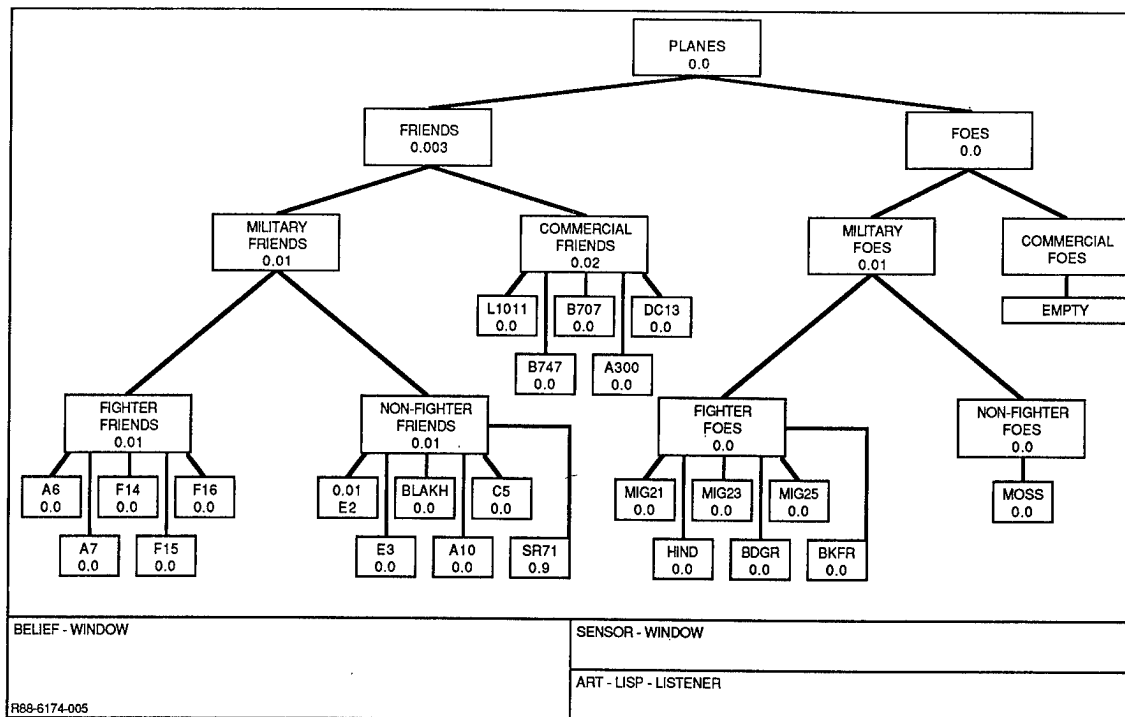


Fig. 5 Evidence Tree

and uses kinematic-discriminant-feature data. The kinematic features included maximal altitude and speed of targets.

The system output, shown in Fig. 5, is in the form of a hierarchical tree with beliefs computed for each node representing one or more targets or target classes. The tree structure represents the frame of discernment based on prior knowledge. The use of prior knowledge allowed for a significant reduction of the power set of possible target combinations to a denumerable linear subset. In this example, the target identified is an SR71 with a belief of 0.9. This result is then used in the tracking ES to support the multitarget multi-sensor track association process and operator decision aiding.

REFERENCES

1. S.S. Blackman, "Multiple-Target Tracking with Radar Applications," Artech House, Dedham, MA (1986).
2. R.M. Yannone, Expert Systems in the Fighter of the 1990s, IEEE Aerospace & Electronic Systems, Vol. I, February 1986, pp. 12-16.
3. P. Bonissone, "Naval Situation Assessment," Proc. of the Uncertainty in Artificial Intelligence Workshop, Seattle, WA, July 1987. Available from AAAI.
4. I. Kadar, "Data Fusion by Perceptual Reasoning & Prediction," Proc. of the 1987 Tri-Service Data Fusion Symposium, Johns Hopkins University Applied Physics Laboratory, Laurel, MD, June 1987.

MULTISENSOR ADAPTIVE CONTROL FOR THE ADI ENGAGEMENT PROCESS

Mac L. Hartless
GE Aerospace
Aerospace Electronic Systems
Utica, New York 13503

ABSTRACT

This paper investigates methods for enhancing acquisition performance for an Atmospheric Defense Initiative (ADI) engagement platform utilizing a multisensor system. A multisensor constant false alarm rate (MSCFAR) processor is described which maximizes overall engagement platform detection probability for the multisensor suite by adaptively controlling each sensor's threshold setting using local background and desired signal estimates while keeping a fixed false alarm rate for the entire multisensor system. A computer simulation is used to provide quantitative results of the benefits of multisensor adaptive control for a system consisting of a radar and a dual band Infrared Search and Track (IRST) system.

The assumptions for the analysis are that the radar target has Swerling III fluctuation and is imbedded in Rayleigh noise. The IR signature of the target is nonfluctuating and is imbedded in Gaussian noise. All spatial and temporal alignment for each sensor's data is assumed to have been already accomplished.

The quantitative results obtained indicate that maximum probability of detection (P_d) enhancement for a multisensor system is achieved when all sensors are operating with similar detection capability. When one sensor has a much higher probability of detecting the target than any of the other sensors, the P_d enhancement using a multisensor CFAR is not significant in itself; however, the information obtained can be useful for optimally managing the sensors via cueing (i.e., dwell time and field of view management) to maximize overall mission success.

MULTISENSOR ADAPTIVE CONTROL FOR THE ADI ENGAGEMENT PROCESS

Mac L. Hartless
GE Aerospace
Aerospace Electronic Systems
Utica, New York 13503

1.0 INTRODUCTION

Avionics performance requirements for next generation ADI weapon systems cannot be met by conventional stand-alone sensor systems. The projected operational requirements include long range target detection, precision multiple target tracking, target classification and identification, and a high level of immunity from countermeasures. No single sensor system will be able to provide all of these capabilities against all types of targets under all operating conditions.

A significant potential performance improvement is available by utilizing complementary sensors such as radar, infrared, and electronic support measures in an integrated manner. To achieve the highest level of performance from a multiple sensor system, it is necessary to do more than merely fuse the outputs of the individual sensor track file outputs. A multisensor system which fully integrates the sensor functions at the signal and data processing levels will provide performance and cost benefits unattainable by any other means.

The Atmospheric Defense Initiative is concerned with CONUS defense against incoming cruise missiles. Due to the low observable nature of the threat and its ability to fly at altitudes which create severe clutter

problems for radar and IR sensors, all sensor systems onboard the aircraft must be utilized in a near-optimal manner to effectively counter the threat.

The ADI mission provides a stressing scenario in which to test some new concepts being developed by GE for the application of multisensor systems. The baseline scenario which will be referred to throughout the paper is the engagement problem consisting of the surveillance platform initially detecting the target and then handing off the target to an engagement platform which is responsible for acquiring the target and counter- ing it.

The focus of this paper will be on the development of adaptive control over all available sensors, and results will be presented on the application of a multisensor constant false alarm rate processor for the engagement platform which will utilize all information previously obtained by the surveillance platform along with its own data to allow optimal target acquisition. The multisensor fire-control architecture used throughout the paper is depicted in Figure 1 which indicates the various sensors which are utilized along with the assumed interaction among subfunctions.

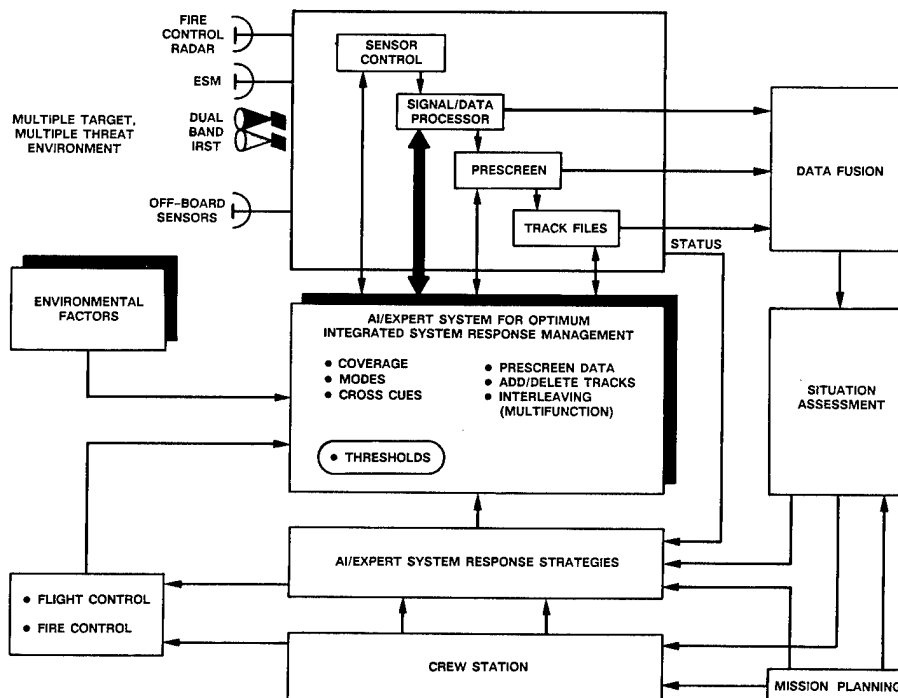


FIGURE 1. FIRE-CONTROL MULTISENSOR ARCHITECTURE

It is assumed that the surveillance platform provides the engagement platform with information about the target state vector and covariance matrix which the engagement platform will use to determine the required field of regard that it must search out to acquire the target. The surveillance platform also supplies information which it has gathered about the target signatures and environmental conditions for all pertinent spectral bands.

2.0 MULTISENSOR ADAPTIVE CONTROL

The term multisensor adaptive control is defined as the controlling function which uses information from other sensors to control some parameter or parameters of another sensor.

A block diagram showing inputs and outputs for the function defined as multisensor adaptive control is shown in Figure 2. The mission inputs which are typically supplied by the overall mission computer determines the priority tasking (i.e., which targets are most threatening versus those targets which are least threatening) and also defines how many false alarms the overall data processing system can handle. The static inputs consist of sensor parameters which do not change quickly with time such as radar power-aperture, frequency bands, and losses. Static variables for an IRST would typically consist of sensor resolution along with the sensor sensitivity in search and track modes assuming an uncluttered blue sky background.

The dynamic variables which may be changing as a function of time are the number of targets to engage, the sensor data inputs, environmental conditions, signature information on the targets of interest, along with target dynamics.

The multisensor adaptive control function would process signal and background information for each sensor cell for which it has information and estimate each cell's signal to interference (SIR) level to optimize sensor suite performance for a variety of different modes such as surveillance, track, or identification capability using the available information from the onboard sensors, along with the constraints imposed by the mission inputs and data processing system capabilities.

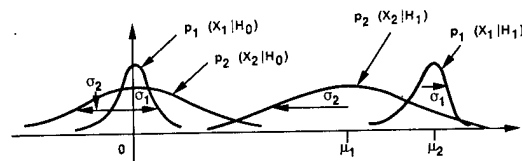
The outputs which would be fed back to the individual sensors consist of threshold multipliers (i.e., the constants which multiply the individual sensor's background estimates), coverage information, timeline management information, and other information concerned with optimal sensor management.

Two examples to illustrate the principle of a multisensor CFAR are presented next. The first example considers two IR sensors with perfect cell alignment and similar statistical behavior. In the second example, a radar with different cell sizes and Swerling III target fluctuation is added to the two IR sensors of the first example to show the effect of combining information from dissimilar sensors.

2.1 Dual Band IR System (Two Sensor Example)

The first example in optimizing multisensor performance in the acquisition mode will be for two IR sensors operating in Gaussian noise with a non-fluctuating target. The target signature's sensor sensitivities are assumed to be different in both bands of interest.

The assumed statistical nature of the two IR sensors' spatial filter outputs is shown below in Figure 3.



THE DENSITIES ARE:

$$x_1 = n_1 : p_1(x_1|H_0) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-x_1^2/2\sigma_1^2} : \text{NO SIGNAL}$$

$$x_1 = n_1 + S : p_1(x_1|H_1) = \frac{1}{\sqrt{2\pi}\sigma_1} e^{-(x_1 - \mu_1)^2/2\sigma_1^2} : \text{SIGNAL PRESENT}$$

$$x_2 = n_2 : p_2(x_2|H_0) = \frac{1}{\sqrt{2\pi}\sigma_2} e^{-x_2^2/2\sigma_2^2} : \text{NO SIGNAL}$$

$$x_2 = n_2 + S : p_2(x_2|H_1) = \frac{1}{\sqrt{2\pi}\sigma_2} e^{-(x_2 - \mu_2)^2/2\sigma_2^2} : \text{SIGNAL PRESENT}$$

FIGURE 3. SENSORS WITH GAUSSIAN NOISE

It is assumed that both IR sensors are viewing the same angular field of view and hence are perfectly correlated in the spatial domain and that the data is taken at close enough instants in time so that the target has not moved from one cell to the next.

The current method of detection processing for two IR sensors is shown in Figure 4. Current dual band IR sensor systems operate autonomously with no adaptive or intelligent system optimizing performance for the overall mission.

The following is an investigation of the treatment of the multisensor suite as a single system with the constraint of a constant false alarm rate for the entire group of sensors, and the optimality criteria of maximum overall detection probability is the quantity which will be maximized.

The overall architecture for the multisensor CFAR is shown in Figure 5. The MSCFAR uses estimates of the signal to interference ratio for each cell and then determines the optimal method of selecting each individual sensor threshold so that a constant overall system probability of false alarm (P_{fa}) can be maintained.

The motivation behind the MSCFAR is that to the author's knowledge, no multisensor system currently in operation combines information from all sensors to optimize performance. A practical example of MSCFAR operation is the case of two sensors arbitrarily set to the same P_{fa} assuming equal performance. In reality, the signature of the target is much higher for one sensor than for the other (i.e., a hot IR target will have a stronger signature in the IR band at the shorter wavelength); therefore, the MSCFAR would raise the P_{fa} (lower the threshold) for the higher signature sensor (maximizing P_d) and lower the P_{fa} (raising the threshold) for the other sensor to maintain constant system P_{fa} .

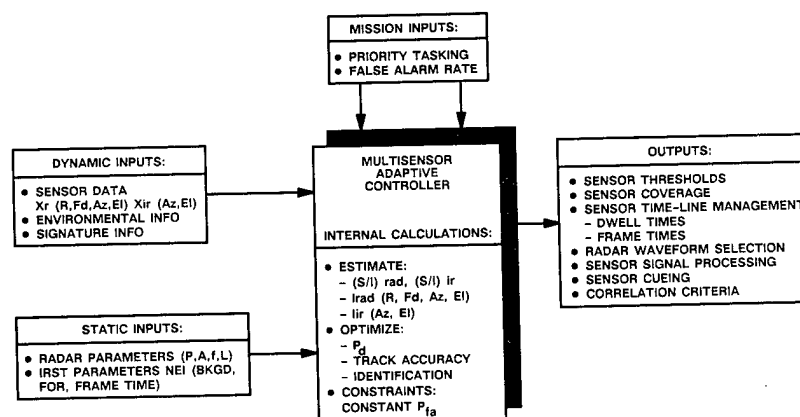


FIGURE 2. MULTISENSOR ADAPTIVE CONTROL FUNCTION

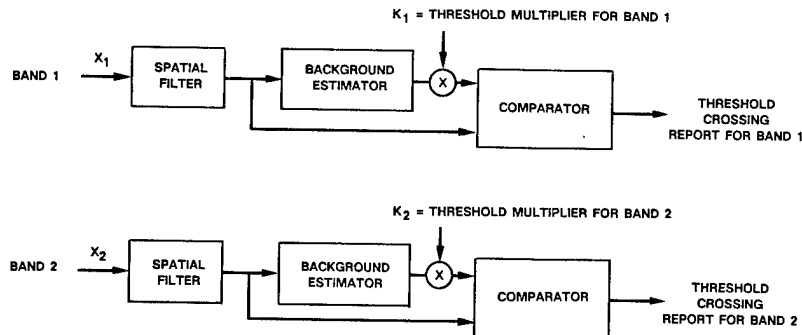


FIGURE 4. CURRENT DUAL BAND IR DETECTION PROCESSING

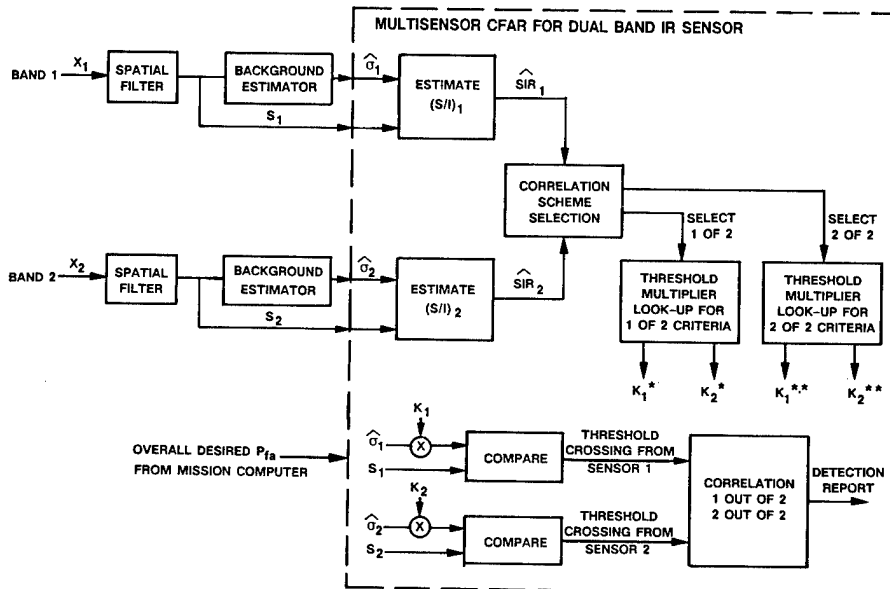


FIGURE 5. MULTISENSOR CFAR CONCEPT FOR DUAL BAND IRST

■ 1 out of 2 (assuming Gaussian noise)

- Maximize: $P_d = 1 - P_{m1} \cdot P_{m2} = 1 - \int_{-\infty}^{\infty} \frac{K_1 \cdot \hat{\sigma}_1}{\sqrt{2\pi} \sigma_1} \frac{-(X-S_1)^2}{2\sigma_1^2} \cdot \int_{-\infty}^{\infty} \frac{K_2 \cdot \hat{\sigma}_2}{\sqrt{2\pi} \sigma_2} \frac{-(Y-S_2)^2}{2\sigma_2^2}$
- Constraint: $P_{fa} = 1 - [(1 - P_{fa1})(1 - P_{fa2})]$

$$\approx P_{fa1} + P_{fa2} = \int_{K_1 \cdot \hat{\sigma}_1}^{\infty} \frac{e^{-\frac{x^2}{2\sigma_1^2}}}{\sqrt{2\pi} \sigma_1} dx + \int_{K_2 \cdot \hat{\sigma}_2}^{\infty} \frac{e^{-\frac{y^2}{2\sigma_2^2}}}{\sqrt{2\pi} \sigma_2} dy$$

■ 2 out of 2

- $P_d = P_{d1} \cdot P_{d2}$
- $P_{fa} = P_{fa1} \cdot P_{fa2}$

• Can Lower Thresholds Considerably When Requiring Two Sensor Correlation

FIGURE 6. TWO SENSOR DETECTION WITH CORRELATION

For the cases where there is a one-to-one cell mapping, as for perfectly aligned IR sensors with the same instantaneous fields of view, the optimal thresholds can be calculated in two different ways. One sensor could be required to have a threshold crossing (i.e., the 1 of 2 criteria), or both sensors could be required to have a threshold crossing for a detection to take place (i.e., the 2 of 2 criteria). It is intuitively obvious that the thresholds can be set much lower when requiring the 2 of 2 criteria while still maintaining the same overall system P_{fa} .

The equations relating the probabilities of detection and the probabilities of false alarm are shown in Figure 6.

To solve for the optimal threshold multipliers K_1 and K_2 , a computer simulation was developed which searches over the constant P_{fa} manifold given by the preceding equations for the threshold multipliers which yield the highest system P_d for the fixed false alarm probability specified for the overall sensor suite.

To get a feel for what the results should look like for the 1 of 2 criteria, the overall system P_{fa} was set to 0.0001 and the estimated SIRs are assumed to be $SIR1=4.0$ and $SIR2 = 5.0$.

Then, as P_{fa1} is varied, P_{fa2} can be solved for directly as:

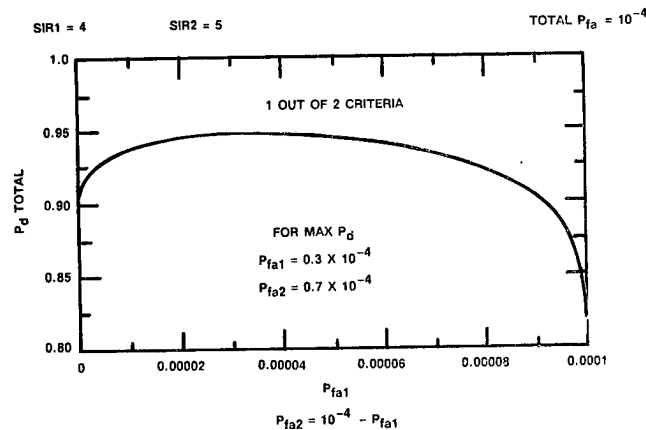


FIGURE 7. TWO SENSOR EXAMPLE

$$P_{fa2} = 0.0001 - P_{fa1}$$

The graph of overall P_d is shown in Figure 7.

Visually one can pick the individual P_{fa} s where maximum P_d occurs to be $P_{fa1} = 0.00003$ and $P_{fa2} = 0.00007$. This makes intuitive sense (i.e., a sensor with higher SIR should be allowed more chance of giving a false alarm since more information is being obtained from that sensor). For this example, it is seen that only 4 percent is gained by choosing the optimal set of thresholds using the 1 of 2 criteria rather than just allowing the dominant sensor (the one with higher estimated SIR) to have all of the P_{fa} .

Including the option of a 2 of 2 criteria selection requires that the optimal P_d be calculated for both criteria and then the criterion resulting in the highest P_d be selected for use.

A representative example using both criteria is shown below in Table 1. This example shows that improvements in detectivity on the order of 13 percent can be achieved when sensors are working at similar detection performances using the 2 of 2 criteria.

By searching over the constant P_{fa} manifolds for both criteria as a function of input SIRs, the selection region for a system requiring an overall P_{fa} of 0.00001 is shown in Figure 8, along with the overall maximum P_d which was achieved.

The following cases shown in Figures 9 and 10 show that the criteria selection region changes as the requirement of overall P_{fa} changes and the 2 of 2 criteria provides better overall system P_d along the region when both sensors are operating at relatively similar SIRs. The basic trend which is shown when the overall system P_{fa} is made smaller is that the 2 of 2 criteria is generally more often used.

2.2 Dual Band IRST and a Radar (3 Sensor Example)

The next example uses the two IR sensors from the last example and a radar which has different statistical properties than the IR sensors and whose cell dimensions do not match perfectly with the two IR sensors.

The statistical properties of the radar are assumed to be Rayleigh noise along with a Swerling III fluctuating target.

Since the radar cell may be larger in the angular dimension than that of the IR sensors along with the additional dimensions of range and Doppler as shown in Figure 11, perfect correlation from a detection cell in radar to one in an IR sensor does not occur. However, one can still ask the question, "How should the MSCFAR choose the optimal thresholds when a threshold crossing is required in at least one of the three sensors in order to declare that a detection occurred?"

Figure 12 indicates how the individual sensor false alarm probabilities and single hit detection probabilities vary as a function of range for a representative set of sensor sensitivities and target signature values. As can be seen from Figure 12, the radar is the dominant sensor due to its fluctuating target at long range, but as range decreases, the IR sensor operating in band 2 begins to detect the target and, since its target is non-fluctuating, its P_d curve has a much faster slope (even though the SNR varies as $1/R^2$ versus $1/R^4$ for the radar), eventually overtaking the radar as the dominant sensor. The overall P_d is seen to be significantly higher than each of the individual sensor's P_d s for cases where the individual sensors have approximately the same P_d ; however, the overall P_d generally tracks the dominant sensor when one sensor provides a much higher P_d than the other two. The latter result is shown better in Figure 13 where the radar still has higher P_d for longer ranges but the IR sensor in band 2 begins to detect the target much earlier than before and generally is the dominant sensor over most of the useful range of the sensor

TABLE 1. TWO SENSOR EXAMPLE

■ Example 1: SNR 1 = 4 SNR 2 = 5 $P_{faTOT} = 0.0001$	
• 1 out of 2 Correlation Criteria: Th1 = 4.0 Th2 = 3.8	
$P_{d1} = 0.49$ $P_{d2} = 0.88$	$P_{dTOT} = 0.94$
	$\Delta P_d = 0.02$
• 2 out of 2 Correlation Criteria: Th1 = 2.0 Th2 = 2.6	
$P_{d1} = 0.97$ $P_{d2} = 0.99$	$P_{dTOT} = 0.96$
■ Example 2: SNR 1 = 3 SNR 2 = 3 $P_{faTOT} = 0.0001$	
• 1 out of 2 Correlation Criteria: Th1 = 3.9 Th2 = 3.9	
$P_{d1} = 0.19$ $P_{d2} = 0.19$	$P_{dTOT} = 0.34$
	$\Delta P_d = 0.13$
• 2 out of 2 Correlation Criteria: Th1 = 2.3 Th2 = 2.3	
$P_{d1} = 0.75$ $P_{d2} = 0.75$	$P_{dTOT} = 0.57$

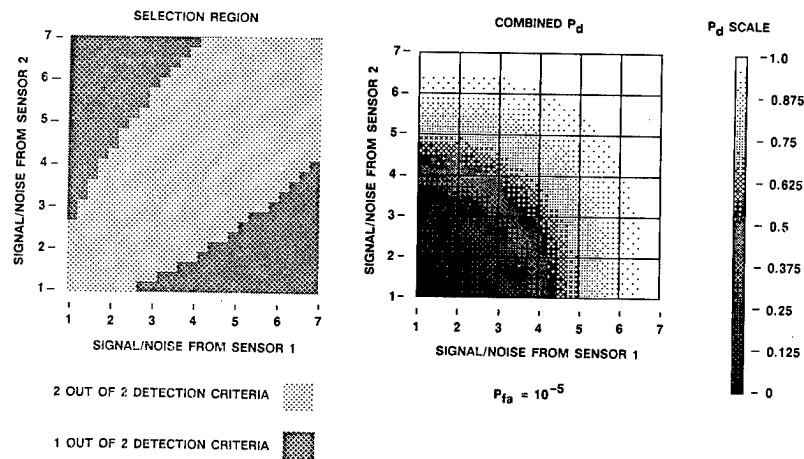


FIGURE 8. TWO SENSOR MAXIMUM PROBABILITY OF DETECTION

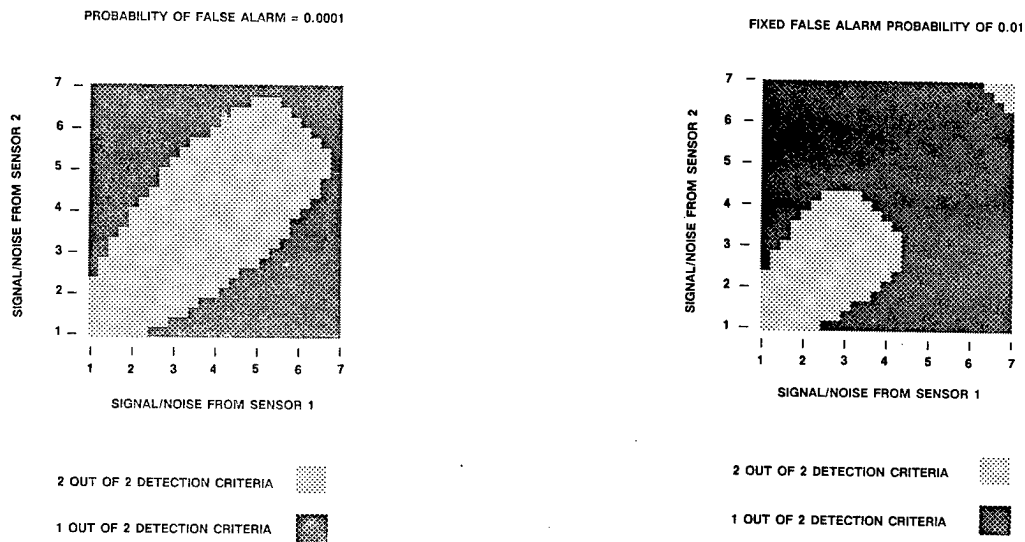
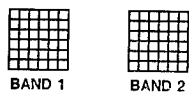


FIGURE 9. TWO SENSOR CORRELATION SCHEME FOR MAXIMUM PROBABILITY OF DETECTION CRITERIA

FIGURE 10. CORRELATION SCHEME FOR MAXIMUM PROBABILITY OF DETECTION CRITERIA

- One-to-one Cell Mapping
 - IR Sensors with identical instantaneous fields-of-view



- Many-to-many Cell Mapping
 - Many IR Cells corresponding to many Radar Cells

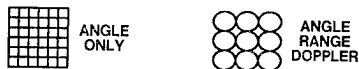


FIGURE 11. SENSOR-TO-SENSOR CORRELATION

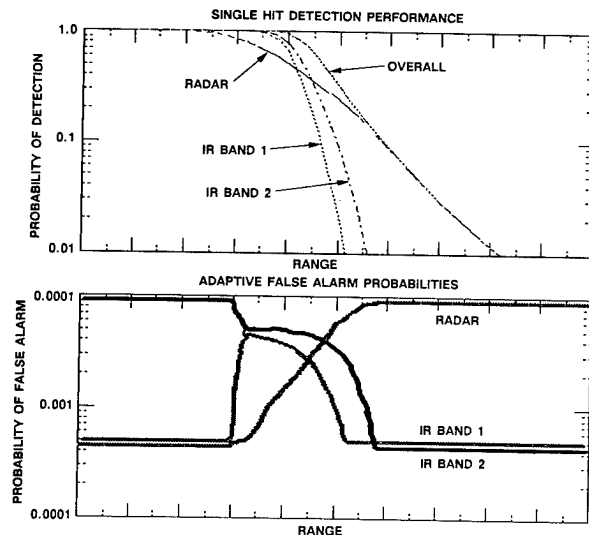


FIGURE 12. MULTISENSOR CFAR EXAMPLE 1

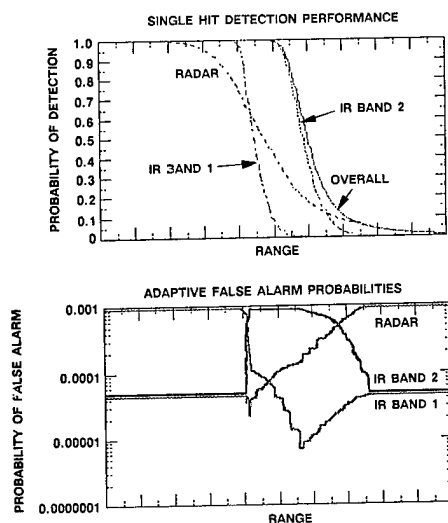


FIGURE 13. MULTISENSOR CFAR EXAMPLE 2

suite. The P_d enhancement in Figure 13 is less than that shown in Figure 12 because over most of the range, one sensor dominated in terms of overall detection capability.

Although the P_d enhancement is not significant in all cases, the information which is obtained from the MSCFAR can be used very effectively for optimal sensor management. As an example, reference is made to Figure 13 where the sensor in IR band 2 dominated in terms of detection performance over most of the range of interest. This condition allows one

to relax the requirement of using the radar to search in that area so that the radar can concentrate its efforts in another region where the IR system may have a clutter problem or severe atmospheric attenuation. The radar can then be cued by theIRST to provide range on the targets in that area where the IR system is providing adequate surveillance. Hence, in addition to optimal thresholds being determined from the multisensor adaptive control function, information is obtained which can be useful for optimally managing the sensors via cueing and sensor management (i.e., dwell time and field-of-view management).

3.0 CONCLUSIONS

As can be seen from Figures 12 and 13, the P_d enhancements obtained using a MSCFAR are limited to regions where at least two sensors are operating with relatively similar performance. Note that this similar performance does not necessarily mean similar sensitivity, but rather similar P_d which is related to a variety of factors (i.e., target signature, target fluctuation statistics, noise statistics, atmospheric and environmental conditions, and sensor sensitivity). All of these factors influence the P_d at the output of the individual sensor's threshold detection logic.

The primary benefit of the MSCFAR is the information that it provides about each individual sensor's performance in a particular region. If the P_{fa} chosen for a particular sensor is considerably lower than the P_{fa} of the other sensors observing the same angular coverage, MSCFAR indicates that this sensor has a very low probability of detecting the target in that coverage area and could be better used looking in an area for which the environmental or clutter conditions allow the sensor to work at its optimal capability.

What has been presented in this paper is a first cut at some of the issues which must be examined for optimal performance of a multisensor system. The examples presented consider how a multisensor constant false alarm rate (MSCFAR) processor might be applied to the case of a dual band IR sensor and a radar for optimal detection performance. Other optimality criteria for tracking and identification need to be developed for effective multifunction operation for the overall multisensor suite.

THIS PAGE LEFT BLANK INTENTIONALLY

Data Representation and Matching for Events and Templates

J.R. Gabriel
Mathematics and Computer Science Division
Argonne National Laboratory Argonne, IL 60439-
4844

M.H. Gabriel
M.H. Gabriel & Associates
680W. 97th St. Lemont, IL 60439

Introduction

This note examines means and reasons to encode event descriptions into bit strings - or, rather, strings of decision flags taking the values TRUE, FALSE, or UNKNOWN. In the case of perfect knowledge, all flags are TRUE or FALSE. In variants of the theory where degrees of confidence are taken into account, UNKNOWN is replaced by a value in the interval $[0:1]$ representing the degree of confidence that the TRUE branch of the decision matches ground truth or the result of perfect knowledge. In templates, a fourth value - ARB - may be used, indicating an event attribute the same as or related to attributes of other events in the same template.

We consider here the encoding of events whose place, time, and other attributes are known. Specifically we show that any set of events so encoded can be sorted to facilitate search for patterns such as troop concentrations. Templates are also sets of events with some attributes UNKNOWN and others ARB in the sense that their values are determined once the values of ARB's for other events in the template are known from matching (unification) with observed events in a history. Thus templates containing an ARB are properly speaking pattern matching algorithms, and can in principle be complex procedures. A match with an UNKNOWN attribute in a template can be TRUE or FALSE and, unlike ARB, does not propagate a value into other attributes.

The possibility of linearly ordering the events in a history and perhaps in a template also enables one to search an event file for aggregates to be done by passing "windows" over the event file.

Decisions About Place in One Dimension.

The one dimensional case, though quite simple, introduces a useful point of view about event files.

Consider a one dimensional Area of Interest (AOI), being simply a line segment whose points are identified by the real numbers in the half open interval $[0:1)$. Suppose we have a detection procedure able to distinguish events in $[0:1/2)$ from those in $[1/2:1)$. If an event is detected in the first subinterval and none in the second, encode it by a 0; if the reverse is true encode it by a 1; if events are present in both segments, mark one to be considered later and encode the other, if no events are present, record the presence of no events.

Next divide the segment containing the event in two equal parts. At least one part must contain an event; if both contain events, mark one for future consideration, and encode the other. Continue until the limit of detector resolving power is reached. One event will have been identified by a bit string recording the decisions, and others may be marked as having been partially identified. If there are marked events still to be processed, these may be encoded by continuing the decision process from where the nearest was saved for later processing, and re-evaluating the part of the decision tree descending from the save-node.

This decision procedure shows how to locate and count events in a one dimensional AOI, and illustrates clearly how the ability to count is limited by the ability to resolve, provided no event attributes other than spatial ones are present. It should be noted that the concept of resolution presented here differs somewhat from the resolution of electromagnetic detectors for example, because there is an unequivocal

cal decision about location of events even when they are close to a boundary where the encoding changes, but which would not be clearly recognised by a real detector. This model of resolution is nevertheless good enough to allow analysis of the effects of resolution on INTEL analysis.

More General Encoding of Position in Space and Time

Consider a spatio-temporal AOI with two space dimensions and one time dimension, i.e. where events take place in a two dimensional region of space (e.g. UTM zone 32U) and in a given time interval (e.g. the calendar year 1981). The two space coordinates x and y , and the time t of an event may each be encoded by binary decision as outlined. If the three codes (x,y,t) are simply concatenated, we have an ordinary Cartesian coordinate system. A more useful encoding for our purposes is to concatenate the leading bits of the x,y,t codes into a tribit code showing which of eight subregions in the AOI contains the event. The indicated subregion may now be further divided by a second tribit code constructed from the next most significant bits of the coordinates. This process may be continued until the precision of the data is completely preserved by the new encoding, and recorded by a concatenation of the tribit codes in the order they were decided, early decisions being in more significant positions in the string. In practice the subdivisions of the AOI are rectangular congruent tiles because this simplifies determination of geographic coordinates. However at the cost of more complex coordinate calculation the subdivisions may be made in any way we please.

These encodings have useful properties. If a set of events, have their positions encoded in this way and the events are sorted so that the encodings are in odometer order, then events where the decisions about spatio-temporal position diverge only late in the decision tree will be close in the file as well as being close in space and time.

The converse is not true however. Consider a four bit encoding of position on a line, the two events (0111) and (1000) are in neighbouring small segments of the line, but are separated at the first decision because one is at the midpoint of the line, and the other slightly nearer to the origin.

In spite of the discontinuities, whose circumvention will be discussed later, spatio-temporal aggregates such as troop concentrations not crossing major decision boundaries will be seen as high densities of events in particular regions of a linear file, and thus can be recognised in a single pass through the file by constructing histograms binned according to high order bit patterns.

The search algorithm is intrinsically parallel and fits well to hypercube architecture. The question of other parallel architectures than MIMD has not been considered properly, but there are no clearly visible difficulties for non MIMD machines.

The event file may show long-range order because for example an aggregate extending across the plane $x=1/2$ in a coordinate system where the sides of the AOI are all normalised to unit length will give rise to two peaks, one with leading bit 1, and the other with leading bit 0. The way the signature develops can be seen most easily in a two dimensional 4x4 example. The encodings for the smallest resolved regions are set out below in an array geometrically similar to the physical AOI.

```
(0101)(0111)(1101)(1111)
(0100)(0110)(1100)(1110)
(0001)(0011)(1001)(1011)
(0000)(0010)(1000)(1010)
```

If bins in a histogram are numbered by the integers corresponding to these bit encodings, a concentration in the four center squares will lead to peaks in the 16 bin histogram at bins 3, 6, 9, and 12. If the concentration is in the upper right, it will be seen in bins 12, 13, 14, and 15. If the concentration is in the right half of the region but centered in the y direction, the peaks will be found in bins 9, 11, 12, and 14. In each case, characteristic peaks will be seen in the histogram, and can be recognised. The rules determining patterns where concentrations cross boundaries arise quite simply from bit inversion in the encoding, for example neighbouring elements across the midpoint of a one dimensional AOI with 16 elements are (0111) and (1000). These are bitwise complementary, the complementarity being a signature for the particular neighborhood relation. Once a search algorithm has recognised a possible troop concentration, the representation lends itself naturally to graphic display for a human analyst.

In our two dimensional case the complete complementarity is seen only for the four center squares. The second case i.e. the concentration in the upper right appears as quite local to the locations (11AA) where A may take the values 0 or 1, and the third case shows complementarity only for y coordinates. Thus if a histogram of events is made at some resolution of interest, local concentrations will be seen as peaks, and concerted action will be seen as long range correlations in bins connected by the complementarity relations.

Adjacency and Proximity

Except for the cases connected by complementarity, proximity of two events is characterised by coincidence of the events at some resolution less than the maximum possible resolution in the representation, i.e. by descent from a common node in the decision tree determining the representation of position. In the case of complementarity, nearby events show complementary paths in the decision trees for one or more coordinates, and identical paths in the others. These relations can be easily tested by masking, inversion, and exclusive OR operations on bitstring representations of position.

Once two decision paths have diverged from regular patterns of either identity or complementarity, the path from one event to the other may be determined by examination of the low-order bits beyond the point of separation, and the distance between the two events determined. Once again, apart from the foldings of the decision tree onto itself induced by complementarity, leaves close in space-time are close in the tree, and leaves close in the tree are always close in space-time.

Other Event Attributes

So far we have been concerned only with spatio-temporal attributes of events. A corresponding binary classification can be introduced for elements of the Table of Organisation (units in the chain of command), and for the Table of Equipment (pieces of equipment). Since the description of equipment is a Directed Acyclic Graph (more than one weapon fires the same 7.62 mm ammunition) the decision process is a little different, but if all multiple choice decisions are reduced to binary ones, bit encodings can still be used.

There are still questions about whether the encodings should be balanced (i.e. whether they are Shannon optimal and arise from divisions into equal classes at each stage), or whether they should follow the natural structure of the graph. Since the natural encoding can always be derived from a Shannon encoding by a suitable computation, this decision depends on a tradeoff between use of CPU time and consumption of file space. The translation to and from optimal Shannon codes can always be hidden from a user by an interface to the database. Thus we can always assume here that natural encodings are used.

In summary, a complete description of an event is either the concatenation of three bit strings, one a spatio-temporal location, one a description of the unit and its state, and the third a description of equipment and its state, or an interleave of the spatio-temporal

string of tribits with bits from the other descriptors, so that the event descriptor is a string of quinta-bits (32-codes).

The 32-codes have the same adjacency and proximity properties as the position codes alone. It is not clear whether the equipment and command descriptors should simply be interleaved, if they are, motorised rifle regiments are likely to well separated in the file from spatially nearby associated units of other types. A possible compromise would be to interleave equipment and command descriptors with low-order bits in the spatio-temporal descriptor, so that each interleaving takes place at a spatial resolution appropriate for the doctrinal distances belonging with the equipment or unit in question.

The question of using these kinds of techniques for study of patterns in radio communication is not discussed here except to note that a bit encoding of position and signal parametrics would show peaks corresponding to nets, and that if parametrics were least significant it might be possible to recognise common nodes of two nets.

Templates and Matching

Introduction of unit and equipment attributes to the encoding opens new questions. It becomes possible to have items in the event database for units whose command attributes are known, but whose position and equipment attributes are not, and vice versa. For example, an M107 175 mm SP gun seen near Dortmund is likely to belong to the Artillery Division of the 1st British Corps of the Army of the Rhine (Isby & Kamps 1985), but without knowledge of unit insignia we do not know if it belongs to the 5th Regiment or the 32nd. If elements of the 5th Regiment have been observed much further away than the doctrinal geographic span of a heavy artillery regiment, there is presumptive evidence that the gun belongs to the 32nd.

These kinds of inference are familiar in automated reasoning, the simplest matching algorithm is called unification where a pair of events, both with unknown elements are combined with a rule saying they are both members of some larger class (Heavy Artillery in the 1st. British Corps of the Army of the Rhine). If two elements are both members of some larger class, then the attributes determining membership of that class must be the same for each, and attributes at lower levels may be mutually exclusive.

In the example "M107 SP gun" and "Near Dortmund" establish membership of Heavy Artillery Division of 1st Corps of BAOR. Doctrine and

geographical separation establish different regiments, and if only two regiments are possible and one is elsewhere, then regiment is determined.

This process propagates knowledge between observations so as to fill gaps using properties of higher level classifications. One natural expression of these is in logic programming (PROLOG), others are in paradigms emphasising properties and inheritance (SCHEME, SMALLTALK, LISP), others yet are the various kinds of expert system shells. Choice of the proper paradigm for a problem can have great influence on speed of matching.

Examples

Moving Target Identification and Stereoscopic Vision

These are two similar problems. In one we wish to detect differences in a scene as perceived by two independent sensors displaced laterally relative to one another. In the other we wish to detect cases where an object has moved since it was last observed. In both cases we assume the object has already been recognised and the report is of an object of some type at some place and time in the case of Moving Target Identification (MTI), and of views of an object recognised as the same in the case of stereoscopic vision.

Consider the MTI problem. A single pass through the event file can eliminate all reports of the same object at the same place but at different times by an exclusive OR and mask test. If it is desired to restrict attention to movement in some interval of time, this too can be done in the same pass. The resulting set of events will have remained sorted so as to maintain the proximity properties of the mapping from space-time into the selected ordered set. Moving target detection is done in two phases. After a resolution has been selected appropriate to the size of movement it is desired to detect, a window is passed across the set to record all cases where successive observations of the same object are close together in the set as moving targets, and then remove them. All other events are connected to each other by complementarity, connected to previously recorded events by complementarity or are observations in error. This case analysis completes the moving target detection.

Stereoscopic Vision

There are few major differences between stereoscopic vision and Moving Target Indication, except that stereoscopic vision often requires consideration of an object as whole, so that individual events are pixels and the recognition of relative dis-

placement in a stereo pair requires comparison of large sets of events each representing many pixels. It is possible to run a cellular automaton as a parallel grid problem "compacting" an object into a single pixel so as to define the position of the object.

The question still needs to be settled by experiment, but the compaction process should lead to positions differing by the same displacement as the original objects. If there is significant difference between two images of the same object in a stereoscopic pair, then the question becomes one of defining object attributes and their weights so as to perform a feature comparison recognising different views of the same object. It will not be treated here.

The Intelligence Database

Intelligence data encompasses only a small part of ground truth. More is usually known about friendly forces and intentions than about the enemy, but substantial inference is needed to derive information necessary to successful operations. This inference is the task of intelligence analysis. One would like to provide automated aids to the analyst for operations at all levels from tactical company and platoon operations to the strategic planning of a campaign. The representation of situation as a set of concisely encoded events is necessary to support automated analysis, therefore the representation given here is a first step towards this goal.

A database of templates can also be represented as histories, and with the availability of CD ROM rotating memory devices, very large databases can be provided, even for systems to be used in tactical planning. However, because plans inevitably depend on terrain as well as disposition of troops, it seems unlikely templates can be predetermined these will depend both on troops and terrain. Terrain data can however be provided on CD ROM, and dispositions of friendly and enemy troops are increasingly obtainable from technical intelligence collection. It therefore seems likely that templates must be held as simulation algorithms, using a representation of knowledge like that discussed here, and parallel computing methods to achieve satisfactory performance in real time.

Certain problems such as recognising aggregates of various kinds and simply warning a commander of their presence seem possible today, and we suggest ways to do this. The next generation of commander's automated assistants should be able to make reasonably sensible command decisions and present them and their simulated consequences to a commander so as to suggest possibilities that might otherwise have gone unnoticed. This may be more important at a tactical level than a strategic one be-

cause such a tool should help inexperienced troops do well in combat, and may be important in the air land battle.

Acknowledgements

One of the authors is indebted to the U.S. Army Intelligence Center and School (a TRADOC mission) for support under MIPR J86-87 between the School and the Argonne National Laboratory. The other author was supported by IR&D funds of M.H. Gabriel & Associates Incorporated, in work to adapt database technology already in use by the company to INTEL problems.

THIS PAGE LEFT BLANK INTENTIONALLY

A Theory of Intelligence Analysis

J. R. Gabriel

Argonne National Laboratory 9700S.
Cass Ave Argonne, IL 60439-4844

M. A. Griesel

Jet Propulsion Laboratory 4800 Oak
Grove Drive, Pasadena, CA 91109

Introduction

Intelligence Analysis is inference from partial knowledge of the real world. Two commonly held views of the process are either that it is statistical, leading to computable probabilities of truth for the various conclusions, or that it is a theory of preference, leading to partial ordering of degrees of belief in the conclusions.

Both views have strengths and weaknesses. The preference models such as fuzzy logic have sound foundations (as discussed by Goodman (1987)). They represent human behaviour in uncertainty well, but because preference models order conclusions only on the basis of degrees of belief in underlying data, they do not quantify questions of expected gains or losses from a decision. The statistical models are able to deal with the question of expected gain, provided the probability of truth of elementary data can be determined.

This paper discusses a statistical model. It began as a theory of intersections between sets of observed events and sets of events in possible state sequences of a model of combat (Griesel 1986, Gabriel 1986). As our understanding developed we realised that a central issue is the extent of knowledge about possible state sequences of a battle or campaign. In an ideal situation all possible sequences are known and the analysis problem is simply one of pattern matching. In practice this is not at all true, one has only an incomplete database of templates and incomplete knowledge of state.

Thus a full theory has two parts: one formulates the ideal model, and the other discusses the additional difficulty arising from incomplete knowledge of templates and its contribution to Type I and Type II statistical errors. At this stage it seems clear that

results of the second part of a theory depend on details of models of combat, detection processes and the like as well as the approximations necessary to make the theory usable on real computer systems.

The methods to perform the detailed studies of the second part are suggested, but owing to the magnitude of the necessary effort they have not been used except for a simple example. Rather we focus on the first part of the theory. It is simple except that in contrast to most current work, it involves consideration of sets of events extending over intervals of space and time.

Representation of Events

Gabriel and Gabriel (1988) have proposed a representation by paths through a binary decision process to classify an event. A perfectly known event is represented by a bit string, where each bit identifies whether the left or right branch of the decision tree was taken at some node corresponding to presence or absence of a property of the event. For locations of events in space and time the representation is a generalisation of a quadcode. It turns out that 96 bits locate a place and time on the earth's surface within a fraction of an inch, and to better than a second in a decade.

Tables of Organisation and Equipment can also be encoded as binary decision trees. Roughly speaking 22 bits identify a soldier or equipment item, and 10 more encode some thousand or so states of readiness or activity.

Thus an event in this model is presence of personnel or equipment in a given interval of space and time and in a given state. A history, or scenario, is a set of these events. Gabriel and Gabriel (1988) show that a history may be sorted so that events close

together in the sorted file are close in space and time. The converse is not quite true, however. Some events close in space and time to a given event are elsewhere in the file, but in regions simply related to the event under consideration. The event file may be thought of as marks on a line. Two representations are useful. In one, all the attributes of time, place, unit, equipment, and readiness are encoded into position, and the mark simply indicates existence of an event. In this representation, because there are a finite number of possible events, the line is divided into the same number of equal segments. Those segments where an event was observed are marked by a 1, the others by a 0. This "superencoding" of events allows histories to be represented as bit strings. It is not used here.

In the other, time and place are encoded as position in the string, and equipment, unit, and readiness are encoded into a symbol. A blank space on the line is also considered a symbol. In this case the history is a set of symbols on the line. This representation transforms a history or scenario into symbols on a line, where cause and effect in the history generate correlations in the sequence on the line. The decision process for position in space-time recursively subdivides the Area of Interest (AOI), so that descent from a common parent node in the decision tree implies containment in a region of space-time defined by the common parent. This containment property is preserved on the line. This fact seems to permit fairly easy recognition of aggregates in space.

The Intelligence Database and System Dynamics

An Intelligence Database is a history of observations of the past, that is, a set of events each being represented by a path through a classifying decision tree. It is incomplete because it does not contain all the relevant past; for example details of enemy planning for the campaign are likely to be unknown. Owing to its set theoretic foundation it is well matched to modern database models and implementations. The Intelligence Analyst's task divides in two parts:-

1. To infer more knowledge of the past (i.e. ground truth) from the database as more information is received about the present.
2. To use available knowledge of ground truth to determine enemy intent and thus to predict enemy action.

Success in both of these depends on knowledge of constraints on the enemy commander. These are primarily terrain, logistics, doctrine and weather. All of them prevent completely arbitrary action by an enemy, and therefore they introduce cor-

relations in the history over space and time. This introduces correlation in the symbols on the linear representation of events. This correlation is the only evidence of enemy intent, provided that all known events are represented in the database.

In addition to being incomplete, the intelligence database is also contaminated by error. It can be viewed as the output of a noisy data channel from ground truth to the analyst. Distinguishing signals (knowledge about ground truth), from noise (ambiguity about meaning of incomplete observations, and deliberate deception by the enemy) needs knowledge of the possible symbol streams that might occur in ground truth. This knowledge allows study of the collection and analysis processes by the methods of Shannon (1948), and determination of their error rates and vulnerabilities.

Using the Database

If the bit string representation of events is used, there is a finite set of possible events, and therefore of possible state sequences. These are an ensemble or ideal database of possible past histories and future developments in the AOI. This ideal database of possibilities is called an ideal template database. In practice it is very large and would therefore be implemented as a set of algorithms, which, when given an observed history return the possible ground truths consistent with the observations, instead of as a tabulation. Practical implementations do not represent all possibilities completely and accurately, but one hopes they are good enough for the purpose at hand.

The Ideal Template Database

Generation by Simulation

In the ideal template database, all candidate scenarios are sets of events containing only TRUE/FALSE binary classification flags. If enemy doctrine were well known, together with statistics of equipment failure, commander's decisions, and the like were understood, this database could be generated by Monte Carlo simulation. One might try to generate such a database for an engagement at the platoon level as a demonstration of principle, although larger scale problems need more resources than should be spent until some small proofs of principle have been obtained.

Generation by Neural Net Algorithms

It also may be possible that a template database could be generated by learning in a neural network. Possible learning algorithms for considera-

tion are Rumelhart's Back Propagation Algorithm (1986), Kohonen's Feature Mapping (1986) and possibly Josin's Categorization Model (1977).

The Matching Process

In the ideal template database all events contain only TRUE/FALSE flags and the database is enormous, but the matching algorithm is simple. If the known observations are found in the past of a template, to within the error of measurement, then the future in that template is a prediction of a possible future in ground truth. A formal theory of measure of set intersections as criteria of match has been given by Griesel (1986).

Since templates extend from the past into the future, even if the past is perfectly known, it will match more than one template, the different futures being determined by future decisions, and other stochastic phenomena. In addition, since the past is usually known only imperfectly, the known past will match templates with different detailed pasts. Divide these templates into groups, each group containing templates having the same detailed past. Different groups have different pasts. Either the present state is different between two groups, or they may be amalgamated because the differing pasts did not affect the present state. This divides the matching templates into groups with significant differences between them.

Thus functionally distinct groups of possible past and future event sequences differ in the present, and there must be events in the present or future able to distinguish the groups. In addition, unobserved past in a group may be considered, and the probability determined that it would have been unobserved given probabilities of detection and the like. As a result at any given time there are groups of possible templates fitting the observations of the past, different groups having different unobserved pasts, and having various computable probabilities of representing complete ground truth.

Evolution in Time and the Commander's Task

If two groups have the same history in the past and are different in the future, there must be a splitting or divergence in the present or future. At the point of divergence there must be a singularity in the equation governing evolution of history with time. Recognising these singularities seems accessible to spectral theory of operators. Clearly decisions by commanders can create singularities, in fact the commander's task is to recognise when a singularity can be generated, and take action to drive evolution along the most advantageous path.

Measures of Performance

The template database is an ensemble of possible histories or scenarios. It can be assumed that the scenarios are equi-probable, if simulation assigns different probabilities to different scenarios, they can be replicated in the ensemble in proportion to their probabilities.

The template ensemble can be sampled to provide a Markov source of possible histories. The samples provide a symbol stream whose rate of information transmission may be calculated (Shannon 1948). A collection and analysis procedure filters the stream, eventually into data for a commander. The commander makes decisions based on the analysis. In a simulation, these decisions may be compared with the decisions that would result from perfect knowledge of the original samples. The relative costs and gains from the two command decision sequences are a measure of performance for the collection and analysis.

If it is desired to evaluate collection and analysis in isolation instead of by their usefulness to commander, they may be performed on a history up to some point in its' evolution, and subsequent evolution compared with prediction by the analysis, the performance of the INTEL process being evaluated by the weighted measure of the intersection between events in the history and events in the prediction by the analysis algorithm, the weights being determined by military science.

A Simple Example

Consider an AOI divided into a 64 by 64 array of squares, and an ensemble defined by the possible arrangements of identical objects, one to a square in

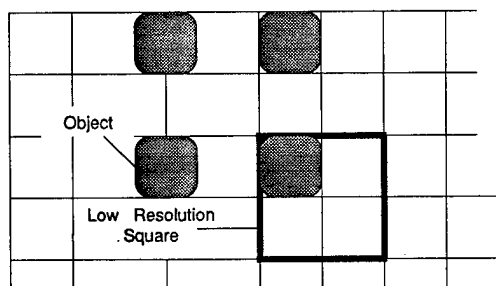


Fig. 1 Example AOI

the AOI (see Fig. 1). An observation of one object by a detector able to resolve coordinate squares yields 12 bits. The resolution of the detector allows all possible configurations to be distinguished, and the corresponding measure of effectiveness is 100%.

If the detector resolution deteriorates by a factor of two in the x and y directions, each observation will yield only 10 bits, and the measure of effectiveness is 25% because only about 1024 of the 4096 possible positions of an object can be distinguished.

Consider the example of a particular configuration of objects, roughly representative of an aggregation problem. Suppose four objects, disposed as shown in Fig. 1. are observed

In the case of adequate resolution the pattern is recognised. If the detector resolution is halved so that a hit by the detector represents from one to four objects in a 2 by 2 array of squares, (as illustrated for the lower right object in Fig. 1.) the detector will report from 1 to 16 objects. The number will depend on how the detector randomly places 2 by 2 squares on the pattern so that each square contains an object, and whether a hit is counted as one, two, three, or four objects in the 2 by 2 square, and so on.

It is perhaps surprising that loss of detector resolution by only a factor of two can cause such severe uncertainties, and if the four objects were farther apart or closer together performance would be better. On the other hand, the configuration shown is typical of an observation early in the development of a troop concentration, and it is therefore a case where good performance has high military value.

Automated Analysis

The Processing Hierarchy

Automated analysis of all observations by one software system at a high level in the command leads to monolithic software and hardware architecture. Such architectures are likely to be difficult to change as necessary improvements are discovered, and, owing to the difficulty of the software development needed in the first place, and unlikely to be properly responsive to real needs. A possible alternative can be based on present manual techniques as discussed in field manuals and the literature on C3I.

The essential feature of these manual methods is distributed hierarchical processing. The first one or two generations of automated systems might be automated versions of the manual process. To design them, a theoretical model is needed allowing measures of performance for both manual and automated technology.

The formal model is a command tree of nodes connected by datapaths, (i.e. a signal flowgraph). The data and material flows should match those in existing C3I. Here we consider only INTEL tasks, and rep-

resent both data and material flows in ground truth by dataflows in the model.

At the root of the tree is the highest command, the leaves are all sensor systems and personnel. Internal nodes are names of organisations rather than equipment or personnel, that is to say the subtree below an internal node describes its parts, only the leaves being physical objects such as people and equipment. In complete C3I model the leaves include maneuver and logistic personnel and equipment, the dataflow includes command and control, and the decision processes include command decisions as well as INTEL tasking and interpretation.

Each internal node of the tree has subordinate nodes corresponding to the command structure. This makes the branching ratio about four or five for most nodes. The datapaths correspond to human comprehension and communication, and thus have roughly the same capacity at each level. At each node there are two decision processes and a local memory.

Incoming data passing through the node towards the root of the tree (the commander) is divided into three streams: one passed on up, one passed to the local store, and one discarded. The local store is purged of obsolete data so as to make space for that coming in.

For dataflow in the opposite direction, orders from higher nodes are augmented by information from the local store before being passed lower in the command structure. The data compression on the way up, and the data expansion on the way down are done by two decision processes, one deciding what to send on, what to store, and what to discard, the other deciding how to augment incoming orders with data from the local store so that they can be processed satisfactorily at the next lower level. The data expansion and compression ratios are the same as the branching ratios for the tree, as they must be to keep the traffic along the datapaths comprehensible by people at the various nodes.

The INTEL decision process for upwards travelling data recognises patterns in incoming information, and classifies them so as to represent the data stream by sequences drawn from a limited number of symbols. This is a Shannon (1948) encoding into a stream of lower capacity. Unless the incoming stream is four or five times redundant some data is lost in the encoding.

In order to recover the essentials of the situation at the lower level, enough of the incoming data is recorded to allow orders travelling downward through

the tree to be expanded so that they can be carried out by lower levels. If saved patterns recur frequently they become permanent templates and eventually affect doctrine. Transient data is discarded as soon as it is too old to be useful.

There is probably a theory of decision about transmission and storage on a C3I network analogous to the theory of neural nets, but the complexity of the decision process may make any practical use of the theory impossible. If the algorithms for decision and data storage have representations by neural nets, then the model is a heterogeneous neural network.

To the extent that the most important data is displayed on overlays at the level where it is used, it seems quite possible that the symbols at each level are not more variable than could be described by existing icons bound to coordinates, with a theory of recognition and value based on unpublished work by Griesel (1986) and Gabriel (1986), using the data representation due to Gabriel and Gabriel (1987).

It is also important to note that as data travels up the tree, the area of interest increases, but the usable spatial resolution changes in proportion so that algorithms to recognise structured aggregates are unlikely to be greatly different at the various levels.

Defining the decision processes at each level by explicit models will be a formidable task. A difficulty in doing this may be the need for transfer of understanding in both directions between intelligence analysts and computer scientists. Proof of principle needs to be established by experiment on a small well defined problem before extensive effort is considered.

Once quantitative models of decision processes are made, it seems a laborious but straightforward task to determine channel bandwidths, and error probabilities using the methods of Shannon (1948; reprint 1963).

This knowledge is important in planning the next generation of automated systems: it gives a quantitative set of Measures of Performance, allowing study of weaknesses of existing systems and proper design of automated replacements. Relevant template models probably exist in simulations now in use, but are likely to need refinement and validation before they are sufficiently reliable for use in automated analysis. It is particularly important in planning to recognise that a satisfactory automated system need not be perfect, it simply must do better than people now do.

Conclusions

There exists a representation of events, suited to recognition of spatial aggregates such as concentrations of troops. Because the representation generates strings of symbols where cause and effect relations will also generate correlations in the symbol stream, the representation is well adapted to discovery of causal connections by a variety of techniques ranging from histograms for spatial aggregates, through parsers for aggregates with bounded context, to algorithms for story understanding in the case of more complex situations.

The representation allows consideration of an ideal template database as an ensemble of possible histories for a campaign. The ensemble model leads to an information theoretic measure of performance for INTEL, and probably for C3I also. Present template implementations typically generate subsets of the ideal ensemble, and could be judged by comparison with the characteristics of the ideal database seen as a Markov source. If a more nearly ideal template database can be generated it is a useful design tool even the generation is costly and slow.

Real template databases, particularly those for tactical use, may be imperfect because they do not contain all possibilities. It seems impossible to know the perfect template database without having perfect simulation. Actual simulation will be imperfect, its imperfections can only be found by comparison with observation of behaviour of our own forces and those of other nations.

An approach to determination of a nearly perfect database is inclusion of patterns seen repeatedly in the past, when their consequences are important. This is a variant on the processes present in neural nets, and thus a possible theory of learning by C3I systems might be taken from theories of learning by neural nets where each neuron is two decision processes and a quite large local store. In this case the problem of predicting changes in enemy doctrine remains open, but if enemy behaviour is seen as a Markov source, changes in behaviour are simply unpredictable changes in the state diagram of the source and the only response is to recognise them as anomalies relative to old behaviour as early as possible.

Failures in this recognition are well documented for the events of August 1914 (Tuchman 1963) and of May 1940 Churchill 1949), and clearly had effects on the history of Europe still affecting us today.

The model of C3I proposed is a hierarchy of decision processes and local memories. This model

matches current manual systems and seems likely to have scale-invariant properties arising from a condition that the data rate of communication between nodes be roughly the same at all levels. This mimics the present manual systems that divide incoming data into summaries and urgent messages to be passed up the tree, and other data to be kept at the node until it is obsolete enough to be safely discarded. It seems likely that the changes of spatial and temporal scale as data passes up the tree might allow use of roughly similar algorithms at all levels to recognise interesting structure. Certainly if current data is always displayed as icons on a screen, then there should be similarities between pattern recognition at the battalion level and at the division level. It is clearly not necessary for C3I automation to be arranged like this, but it would support a gradual transition from manual methods to automated ones, and would allow a limited manual capability in the event of failure of automated systems (except loss of communication links).

Finally a few more philosophical remarks may be helpful in giving perspective about the problem. INTEL analysis is like scientific research in the sense that both are efforts to predict behaviour of the real world from limited observations. The significant difference is that an established scientific truth does not change, although better measurements may refine a theory. In contrast understanding of an enemy may become obsolete because of deliberate enemy decisions, for example change of an encryption, or development of new doctrine.

The history of science may provide useful warnings, a scale for the possible improvement in INTEL technology, and a scale for its' probable difficulty.

Secondly, as pointed out by Shannon (1949), the theory of cryptanalysis as practiced between 1930 and perhaps 1960 is a theory of extracting signals from noise, and at least for the rotor machines, a theory of semi-groups. Simulations evolving in time are also representations of continuous semi-groups. Perhaps some of the unpublished work, particularly from 1940 on, might be of use in INTEL analysis.

Acknowledgements

Particular thanks are due to the U. S. Army Intelligence Center and School, Ft. Huachuca; a TRADOC mission, for support under MIPR J86-87 in work related to this paper, and encouragement to consider the problem. The work began as two position papers, one by each author, presented at a meeting between Argonne and JPL, called in December 1986 by the Scientific Advisor, Ft. Huachuca as part of a Director, Combat Developments initiative.

The author directly responsible for this text is also most grateful to Dr. G. W. Pieper of MCS division at Argonne for her helpful criticisms as an editor. Responsibility for remaining errors of fact or style lies with this author, there would have been many more without Dr. Pieper's help.

References

Churchill, W. S., *The Second World War*; Houghton Mifflin 1949 (regularly reprinted).

Griesel M. A., *Decision Making in All Source Processing*; unpublished information, Dec 1986

Gabriel J. R., *Battlefield Grammars*; unpublished information, Dec 1986

Goodman I. R., *A General Theory for the Fusion of Data*; 1987 Tri Service Data Fusion Symposium Vol 1 9th-11th June 1987, pp 252-270, Naval Air Development Center, Warminster, Pennsylvania

Gabriel J. R. and Gabriel M. H., *Data Representation and Matching for Events and Templates*; Tri-Service Data Fusion Symposium 1988

Josin G., *Self Control in Neural Networks*; *Biological Cybernetics*, Vol 27, 1977

Kohonen T., *Clustering, Taxonomy, and Topological Maps of Patterns*; Sixth International Conference on Pattern Recognition, pp 114-128, Pub IEEE 1986

Rumelhart, McClelland, et al; *Parallel Distributed Processing - Vol. I: Foundations*, M.I.T. Press, 1986

Shannon C.E., *The Mathematical Theory of Communication*; Bell System Technical Journal July and October 1948. Reprinted by University of Illinois Press Urbana, Chicago & London 1963 ISBN 0-252-72548-4

Tuchman B. *The Guns of August*; 1963.

PREDICTIVE MODELS OF CORRELATOR/TRACKER ALGORITHM PERFORMANCE
IN THE PRESENCE OF FALSE ALARMS

Barry Belkin
Gail A. Schweiter
Roberta S. Wenocur

Daniel H. Wagner, Associates
Station Square One
Paoli, Pennsylvania 19301

Our interest is in assessing the performance of a multiple correlation hypothesis correlator/tracker algorithm (CTA) in a false alarm environment. Several methods of analysis are possible.

At one extreme are empirical approaches. For example, one might model the behavior of a CTA when false alarms are absent (an analytically tractable problem) and then hope to discover some form of empirically derived adjustment or scaling law to extrapolate to the case involving false alarms.

At the other end of the spectrum, one might attempt to develop a purely analytic model of a CTA, involving only closed form expressions or possibly recursive relations. Such a model could then be used in parametric studies of the effects of false alarms on CTA performance. One such model is described in "An Analytic Model for the Effect of False Reports on Surveillance Tracking," by Barry Belkin, and Roberta S. Wenocur, *Technical Proceedings of the 1987 Tri-Service Data Fusion Symposium*, Vol. 1, June 1987, pp. 631-640. Unfortunately, the problem of modeling CTA performance analytically is a very difficult one and such models require a great many simplifying assumptions and have very limited predictive capability.

A compromise approach is a hybrid model combining analytic expressions for tractable parts of the problem with Monte Carlo simulation on a very limited scale.

We briefly survey some recent results based on the three types of approaches described.

PREDICTIVE MODELS OF CORRELATOR/TRACKER ALGORITHM PERFORMANCE
IN THE PRESENCE OF FALSE ALARMS

Barry Belkin
Gail A. Schweiter
Roberta S. Wenocur

Daniel H. Wagner, Associates
Station Square One
Paoli, Pennsylvania 19301

1. INTRODUCTION

In multi-target surveillance problems, it is generally not possible to associate each sensor report uniquely with its target of origin. As a result, there is considerable interest in probabilistic methods for report-to-track correlation. One class of such data correlation methods is based on the processing of multiple report-to-track correlation hypotheses in parallel. For example, in reference 1, multiple hypothesis techniques referred to as *track splitting* and *optimal Bayesian* are described. The appeal of multiple hypothesis techniques is that they provide the most accurate probabilistic representation of the target tracks that can be extracted from the sensor data. On the other hand, because of the tendency for the number of correlation hypotheses to grow at an exponential rate, multiple hypothesis methods are generally quite computation intensive.

Continuing advances in computer design, however, are increasingly favoring the use of multiple hypothesis methods. See, for example, reference 2 for a discussion of a multi-hypothesis correlation and tracking algorithm (CTA) for use in the midcourse phase of the Strategic Defense Initiative application.

Our present interest is not so much in multiple hypothesis CTAs themselves, but rather in predictive low-level models of CTAs which serve to characterize CTA performance without formally carrying out the computation-intensive CTA functions. Such models, if available, would find application for example in parametric studies where one is interested in investigating system performance over a wide range of combinations of surveillance problem parameters. The use of an actual multi-hypothesis CTA to carry out such parametric studies would most likely involve a prohibitive amount of computation.

A second application of low-level CTA models is in the study of properties of CTAs which are particularly difficult to investigate computationally. For example, one might be interested in characterizing the problem conditions under which a CTA exhibits a well-defined steady state behavior. It might require many tracking problem replications, each simulating an extended period of real-time processing to obtain statistically meaningful computational results about the steady state. A low-level model, on the other hand, might provide a direct characterization of the steady state allowing one to bypass lengthy computation.

We consider low-level predictive models falling into three general categories: analytic models, empirical models, and simplified CTAs. Analytic models allow one to predict CTA performance from first principles based on closed-form mathematical expressions or recursive relations. Empirical models incorporate estimation techniques such as curve fitting to allow one to extrapolate from a relatively small data base of representative results obtained by running an actual CTA to arbitrary cases. Other than in the curve fitting phase for an empirical model, the analytic and empirical methods avoid the use of an actual CTA in obtaining predictions of CTA performance. The third approach is to use simplified CTAs directly. Since the generation, processing, and pruning of correlation hypotheses are the most computation-intensive aspects of a CTA, the critical simplification of such algorithms involves a sharp reduction in the number of such hypotheses that are considered.

In Section 2, we develop an analytic model for the special case in which there is a single valid target and no

false targets; i.e., the false report density $\lambda_F = 0$. We will cite specific examples of each type of low-level model in the more general case of $\lambda_F > 0$. In Section 3, we will present numerical results summarizing a study of the predictive performance of one particular empirical model based on a so-called *scaling law* and one particular simplified CTA. The actual CTA used in this comparative analysis is the False Alarm Correlation Evaluation Tool (FACET), developed to study the correlation and tracking problem in an environment involving a single valid target and a moderate to high false report density. Conclusions are summarized in Section 4.

2. DISCUSSION OF MODELS AND PARAMETERS

2.1 Low-level models

We consider each of the three broad categories of low-level CTA models enumerated above.

Analytic models. As previously noted, an analytic model of a CTA is characterized by the availability of closed form mathematical expressions for the key output quantities measuring the degree of target localization. Such a model could be used, for example, in parametric studies of the effect of false alarms on CTA performance. An analytic CTA model is described in reference 3. A low-level model for a multiple hypothesis CTA using a maneuvering target model is postulated and an analytic expression is derived for the expected surveillance area that must be searched to localize the valid target. If one is interested in probability distributions rather than expected values, the problem appears to be intractable. In the special case of discrete glimpse opportunities on a single valid target with no false alarms, however, under appropriate assumptions, one can derive a formula for the target localization distribution, i.e., the probability of target containment as a function of area searched.

For several reasons, this baseline case should be studied before introducing the complications of false alarms into any low-level model. First, its relative simplicity allows for direct analysis; second, it provides guidelines for the more challenging situation involving false alarms; third, it is the foundation for models which extrapolate or scale from the no false alarm case.

We select as our performance measure, the valid target localization distribution. Two types of search plans are considered: fixed area search and fixed threshold search.

Fixed area search. Suppose that a fixed search area A is assigned and the problem is to determine the containment probability $P_C(A)$ for this value of area and for false alarm density $\lambda_F = 0$. In this case, a region of area A is chosen to maximize the probability of target containment.

Our approach is to condition first on time elapsed since the last detection of the valid target. For this purpose, we must consider the following variables:

T = time between successive discrete glimpse opportunities

M = number of discrete glimpse opportunities since most recent detection of valid target

t_p = sensor report processing delay time

$S = (t - t_p) \text{ [mod } T]$.

We assume that evaluation takes place at a random time t with respect to the discrete glimpse process. Thus, time t can assume real values other than integer multiples of T . Consequently, S is the time between t and the most recent glimpse opportunity. The total elapsed time since the last detection denoted T_E can be represented as

$$T_E = M \cdot T + S + t_p.$$

We note that M , t , S , and T_E are random variables, while T and t_p are parameters.

Next, we introduce

(t) = Area size of valid target containment region at time t since last detection

and

$\sigma(t)$ = standard deviation of the target location distribution at time t since last detection.

The datum growth law for (t) and the formula for $\sigma(t)$ depend upon the state of knowledge of initial target position and velocity. For the various growth law formulas, see reference 4.

We further assume that the target at any given time is in one of two possible detection states: detectable or undetectable, with rate parameters λ_U and λ_D associated with the exponential distributions for time spent in the undetectable and detectable states, respectively.

For each fixed value m of M , we condition on values of S . We assume that S is uniformly distributed on the interval $[0, T]$, which leads to the conditional containment probability

$$P_C[A|M=m] = \frac{1}{T} \int_0^T \Pr\{\mathcal{A}(mT+s+t_p) \leq A\} ds. \quad (1)$$

Based upon properties of circular bivariate Gaussian distributions and of Integrated Ornstein-Uhlenbeck model (see reference 4) for target motion, the integrand in equation (1) can be expressed as

$$\Pr\{\mathcal{A}(t_E) \leq A\} = 1 - e^{-A/2\pi\sigma^2(t_E)}, \quad (2)$$

where

$$t_E = mT + s + t_p.$$

To condition on values of M , we define

$$P_m = \Pr\{M=m\}.$$

Defining

$$Q_1(t) = \frac{\lambda_U}{\lambda_D + \lambda_U} + \frac{\lambda_D}{\lambda_D + \lambda_U} e^{-(\lambda_D + \lambda_U)t}$$

and

$$Q_0(t) = \frac{\lambda_D}{\lambda_D + \lambda_U} + \frac{\lambda_U}{\lambda_D + \lambda_U} e^{-(\lambda_D + \lambda_U)t},$$

we can now compute probabilities $P_1^{(n)}$, $P_0^{(n)}$, and $P^{(n)}$ using a set of recursive relationships as follows, where P_D is the single glimpse valid target detection probability:

Recursion:

$$\begin{cases} P_1^{(n+1)} = P_1^{(n)} Q_1(T) (1 - P_D) + P_0^{(n)} (1 - Q_0(T)) (1 - P_D) \\ P_0^{(n+1)} = P_1^{(n)} [1 - Q_1(T)] + P_0^{(n)} Q_0(T) \end{cases}$$

and

$$P^{(n)} = P_0^{(n)} + P_1^{(n)}.$$

Initial conditions:

$$\begin{cases} P_1^{(0)} = 1 \\ P_0^{(0)} = 0. \end{cases}$$

Then

$$P_m = \frac{\lambda_U}{\lambda_U + \lambda_D} P_D \cdot P^{(m)}.$$

Conditioning on values m of M , we obtain

$$P_C(A) = \sum_{m=0}^{\infty} P_m P_C[A|M=m],$$

with $P_C[A|M=m]$ defined by equation (1). It is interesting to note that when $\lambda_D = 0$, the target is continuously detectable and P_m reduces to

$$P_m = P_D (1 - P_D)^m,$$

that is, M has a geometric distribution as one would expect.

Fixed threshold search. Instead of assigning a fixed search area, suppose we set a fixed target density threshold ℓ . In principle, this means that search is applied to regions R_ℓ of the plane that satisfy

$$R_\ell = \{(x, y) : f_{t_E}(x, y) \geq \ell\},$$

where $f_{t_E}(x, y)$ is the bivariate Gaussian density for target location that depends upon the datum growth time t_E . These regions are in fact determined by elliptical

equidensity contours of the bivariate Gaussian density.

Let $P_{\max}(t_E)$ be the maximum value attained by a circular bivariate Gaussian density with variance parameter $\sigma^2(t_E)$ given by

$$P_{\max}(t_E) = \frac{1}{2\pi\sigma^2(t_E)}.$$

For each fixed threshold ℓ , one determines which values t_E of elapsed time T_E are meaningful. If t_E is too large, $\sigma(t_E)$ will define a circular bivariate Gaussian density whose maximum value $P_{\max}(t_E)$ is less than ℓ . Conditioning on S and M , and accounting for the two-state Markov detection process by means of P_m one may compute average area $\bar{A}(\ell)$ and corresponding average containment probability $\bar{P}_C(\ell)$, for the given threshold ℓ .

To determine the containment area $A(\ell; t_E)$ corresponding to fixed target density threshold ℓ and elapsed time t_E , set the bivariate circular Gaussian density equal to ℓ and solve for $A(\ell; t_E)$. This yields

$$A(\ell; t_E) = -2\pi\sigma^2(t_E) \ln(2\pi\ell\sigma^2(t_E)). \quad (3)$$

With the threshold ℓ fixed, and for each value m of M , we condition on values of S . As in the fixed area case, we assume that S is uniformly distributed on the interval $[0, T]$, which leads to the conditional containment area

$$A(\ell|M=m) = \frac{1}{T} \int_0^T A(\ell; mT + s + t_p) ds \quad (4)$$

and corresponding conditional containment probability

$$P_C(\ell|M=m) = \frac{1}{T} \int_0^T \Pr\{A(t_E) \leq A(\ell; t_E)\} ds, \quad (5)$$

where once again

$$t_E = mT + s + t_p.$$

By substituting from equation (3) into equation (2), the integrand in equation (5) can be expressed as

$$\Pr\{A(t_E) \leq A(\ell; t_E)\} = 1 - 2\pi\sigma^2(t_E)\ell,$$

which leads to a simple expression for equation (5) for each form that $\sigma^2(t_E)$ assumes.

Conditioning on values of M , we obtain

$$\bar{A}(\ell) = \sum_m P_m A(\ell|M=m) \quad (6)$$

and

$$\bar{P}_C(\ell) = \sum_m P_m P_C(\ell|M=m), \quad (7)$$

where the sums are over all values of m that are meaningful for threshold ℓ , and with $A(\ell|M=m)$ and $P_C(\ell|M=m)$ defined by equation (4) and equation (5), respectively.

We observe that one can show that fixed threshold target containment probability curves uniformly dominate those based upon fixed area search (see reference 5).

Unfortunately, when false alarms are present and multiple correlation hypotheses develop, the problem of determining the target containment distribution appears to be analytically intractable.

Empirical models. In contrast to an analytic model, an empirical model uses the results of actually running a CTA. For example, one might model the behavior of a CTA when false alarms are absent, which leads to an analytically tractable problem, and then attempt to discover some form of empirically derived adjustment or *scaling law* to extrapolate to the case involving false alarms. The basic premise of the scaling law in reference 6 is that any surveillance scenario including false contacts is equivalent in terms of valid target localization statistics to a scenario without false contacts and with a lower valid report rate. The particular adjustment factor is derived empirically by curve fitting to data based on runs of a CTA. The obvious advantage to a scaling law is that once the empirical law is derived, further runs of a CTA become unnecessary.

By their nature, both analytic and empirical models do not allow one to study the computational properties of a CTA. Further, a scaling law provides no information on the region of stability (existence of a steady state) of a CTA.

Simplified CTAs. In part to deal with the above stated limitations of analytic and empirical CTA models, the authors have developed a simplified CTA that combines both analytic and Monte Carlo simulation methods to predict CTA performance. Relative scoring of correlation hypotheses (*scenes*) is based on a Bayesian formulation and false alarms are simulated using a spatial Poisson process. Current scenes condition on past sensor report assignments only as far back as the second most recent valid report; consequently, fewer hypotheses are involved. Type I and Type II errors are of order at most one, where in the present context a Type I error occurs when a valid report is declared false and a Type II error is made when a false report is declared valid. The inter-arrival time distribution for valid detections is derived analytically by assuming the same two-state target detectability process used in the analytic model. Spatial overlap in the conditional target location densities is ignored. This simplified CTA model reproduces many of the operating characteristics of a full-scale CTA but at a higher computational cost than analytic and empirical models.

2.2 Regions of Parameter Space

One can think of surveillance parameter space as being subdivided into three general regions:

- (1) **Region 1:** This is the locus of points in parameter space where the false alarm rate λ_F is low and the predictive performance of low-level models is good to excellent.
- (2) **Region 2:** In this region, λ_F has increased to the point at which the full power of multiple hypothesis CTA logic is being invoked, and the predictive capability of low-level models is generally only fair.
- (3) **Region 3:** This is the region where λ_F has increased to the point that the CTA is saturated with false alarms, tracking in any conventional sense has broken down, and low-level models have no useful predictive value.

It should be pointed out that in Region 3 one could not reasonably expect much from a low-level model since the CTA being modeled is itself past the point of providing any useful tracking capability.

An important issue is the characterization and, in fact, the proof of the existence of the region 3 as defined above. One might reasonably conjecture that as the false report density increases with all other problem parameters remaining fixed, the performance of a CTA simply degrades more or less continuously with no sharp transition between stable and unstable behavior. Interestingly, evidence that this is not the case is reported in Chapter 8 of reference 1.

Specifically, the authors consider what they call the probabilistic data association filter (PDAF). Under the PDAF method, a scan of sensor reports, at most one of which can be on the valid target, is processed as an equivalent single report. Each report in the scan contributes to this equivalent report in proportion to a computed likelihood that the given report is actually valid. The asymptotic properties of the PDAF are investigated by numerically iterating the covariance update equation (sometimes called the Ricatti equation) until the filter output either stabilizes to some type of steady state, or exhibits some form of instability, i.e., the iterates of the Ricatti equation do not converge to an identifiable limit.

Figure 8.2 of reference 1 shows a marked boundary between the region of stability of the PDAF and the region of instability. Because the PDAF is essentially a single hypothesis filter, the existence of these distinct regions does not depend on a choice of scene retention threshold settings used in multiple correlation hypothesis pruning.

Consequently, the methods of reference 1 can be used to investigate how a CTA, even if operating with effectively infinite computing resources, would become unstable if the false report density were high enough. Unfortunately, it is difficult to investigate this instability phenomenon using a true multi-hypothesis CTA such as FACET since an apparent instability may occur because one has set the scene retention thresholds too restrictively.

We observe that it would be valuable to identify a simple quantity that measures the intrinsic difficulty of a tracking problem at least to the point of determining within which region of parameter space a given problem falls. Once one determines the appropriate region for a tracking problem, one can decide whether a low-level model is appropriate. The scaling law described in reference 6 implies the existence of one dimensionless intrinsic quantity of this type. It remains a subject for further investigation whether this quantity is related to the boundaries between the regions in parameter space.

3. RESULTS

Figures 1 and 2 show a direct comparison of FACET output with that of the analytic model described in Section 2. One sees that this model provides an excellent approximation to the output of FACET in the case of no false alarms.

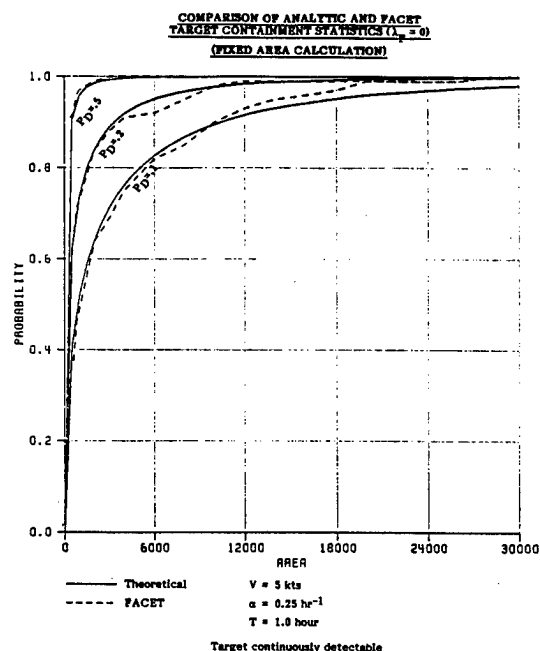


FIGURE 1

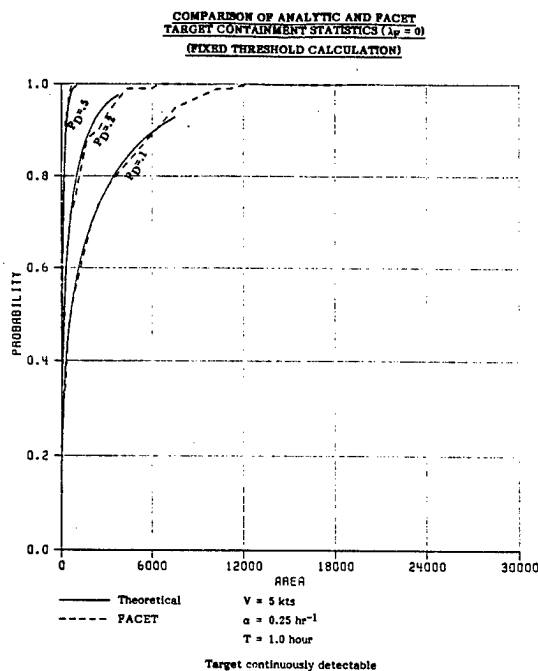


FIGURE 2

Figures 3 through 5 compare FACET output with that of the scaling law for representative parameter values. In Figure 3, tracking is relatively easy and the results are in excellent agreement. In Figure 4, tracking is more difficult and the scaling law underestimates, for a given search area, the probability of localizing the target. In Figure 5, tracking becomes even more difficult but now the scaling law overestimates target containment probability. Additional cases of FACET have been run where there are large discrepancies between FACET and the scaling law; however, there is evidence (reference 7) that the scene retention thresholds in FACET were set too restrictively, though the problem may simply be in Region 3 of parameter space.

Figures 6 through 8 show a comparison of FACET with the simplified CTA as described in Section 2.1 with the same parameter choices as in Figures 3 through 5. The same general trend of tracking of the valid target becoming more difficult is observed, with the simplified CTA more consistently overestimating target containment probability.

COMPARISON OF SCALING LAW AND FACET
TARGET CONTAINMENT STATISTICS

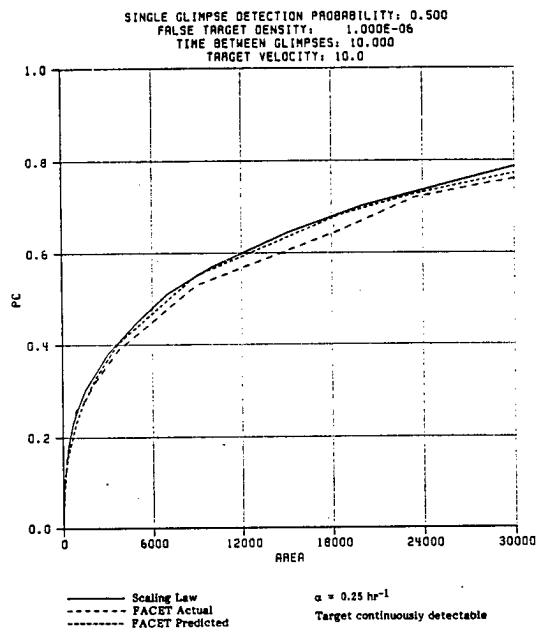


FIGURE 3

COMPARISON OF SCALING LAW AND FACET
TARGET CONTAINMENT STATISTICS

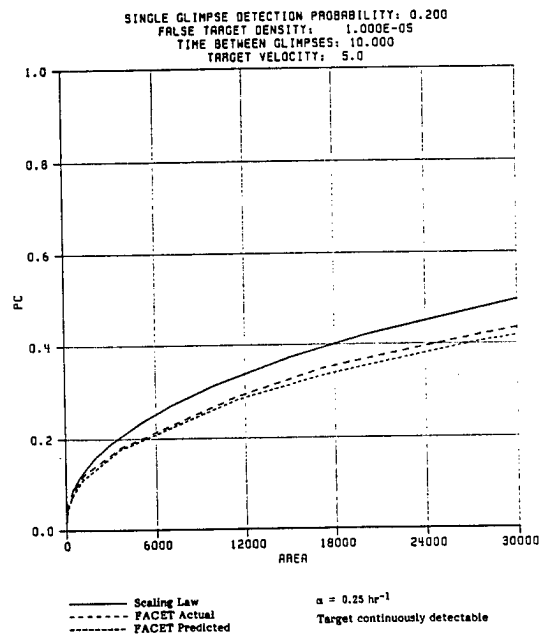


FIGURE 5

COMPARISON OF SCALING LAW AND FACET
TARGET CONTAINMENT STATISTICS

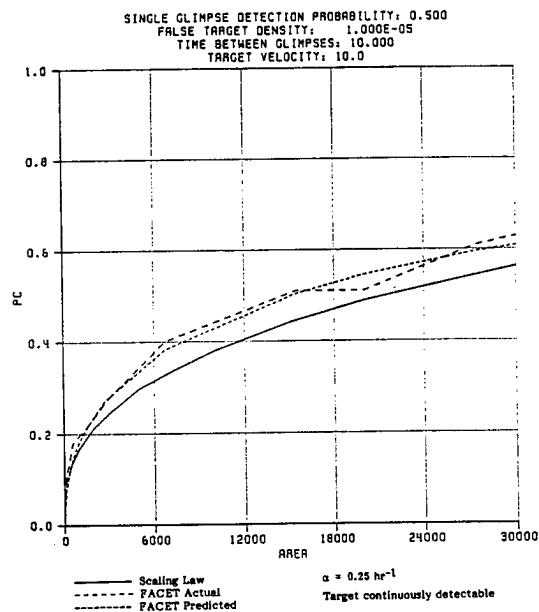


FIGURE 4

COMPARISON OF SEMI-ANALYTIC AND FACET
TARGET CONTAINMENT STATISTICS

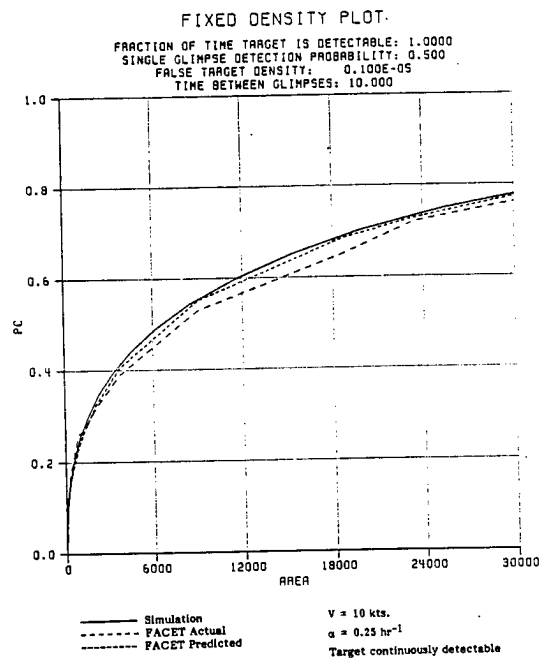


FIGURE 6

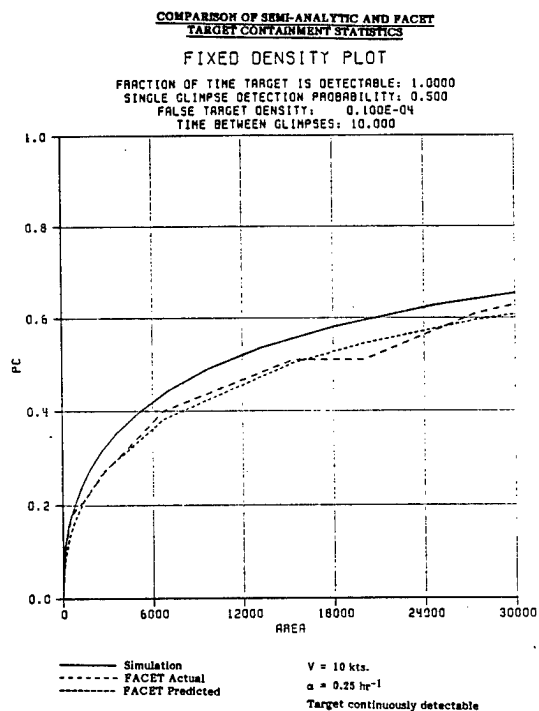


FIGURE 7

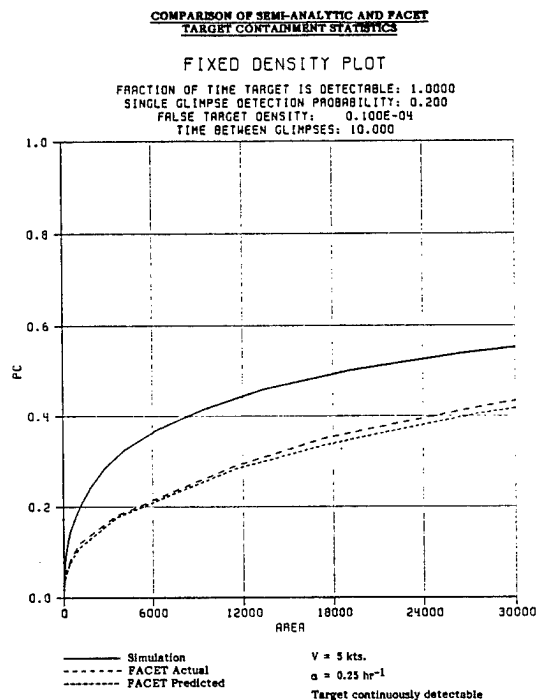


FIGURE 8

4. CONCLUSIONS

The representative CTA FACET and each of the low-level models: the scaling law, the analytic model, and the simplified CTA agree very closely in their estimates of target containment probability versus search area in the special case of no false alarms. When the false alarm density λ_F is small, the scaling law and simplified CTA models continue to agree well with FACET. The problem of finding a purely analytic model in the $\lambda_F > 0$ case appears intractable. As λ_F increases, a growing disparity develops between FACET and the low-level models. Eventually, for λ_F sufficiently large, tracking breaks down totally. Additional studies are needed to determine the boundaries of the regions of parameter space where low-level models show fundamentally different levels of predictive performance.

5. REFERENCES

1. Yaakov Bar-Shalom and Thomas E. Fortmann, "Tracking and Data Association," Vol. 179 in Mathematics in Science and Engineering Series, Academic Press, Orlando, 1988.
2. K. Askin, NRL; J. Erbacher, VERAC, Inc.; and B. Belkin and D. P. Kierstead, DHWA; "SDI Midcourse Tracker/Correlator Algorithm Development," 1987 Tri-Service Data Fusion Symposium Technical Proceedings, Vol. 1, 9-11 June 1987.
3. B. Belkin and R. S. Wenocur, "An Analytic Model for the Effect of False Reports on Surveillance Tracking," 1987 Tri-Service Data Fusion Symposium Technical Proceedings, Vol. 1, 9-11 June 1987.
4. B. Belkin, "The Ornstein-Uhlenbeck Displacement Process as a Model for Target Motion," Daniel H. Wagner, Associates Interim Memorandum to Applied Physics Laboratory, February 1, 1978.
5. E. P. Loane, "Surge Allocation," EPL Analysis Report, no date.
6. E. P. Loane, "Scaling Law Methodology for Surveillance and Surge, with Periodic Glimpses and Observation of Position and Velocity," EPL Analysis Memorandum to Albert J. Cole, October 12, 1987.
7. Ray Jacobovitz, "Comparison of XTRACK, FACET, EPLA Scaling Law," METRON Briefing, April 13, 1988.

DATA ASSOCIATION AND FUSION IN AN ASW ENVIRONMENT

Kenneth McPhillips and Robert Fagan

Raytheon Company
Submarine Signal Division
1847 West Main Road
Portsmouth, RI 02871-1087

This paper addresses the multi-sensor, multi-target data association and data fusion problem for a contact management function in a submarine combat control systems environment. Only geometric mutual data association/fusion is considered and simulated results presented. Data from multiple acoustic sensors are associated at the track level using a χ^2 test. The sensor data can be fused using a nonlinear least squares batch processor; or, the track (solution) data can be fused to create a composite track. The paper discusses the different levels of data fusion: at the measurement level or the track level; and, presents the advantages and disadvantages of each approach. Error types (Type I and II) that are characteristic of an association decision are discussed and experimental (simulated data) results are presented for particular scenarios involving two different platforms. The simulation results show that Type II errors (bad associations) are a function of the scenario and can only be determined experimentally for each scenario. Even manual data association is considered to determine the feasibility of an operator being able to decide whether two different contacts should be associated. The problem of fusing track level data that is produced from different target motion analysis algorithms is examined. Lastly, since the work presented in this paper is of a preliminary nature, data association/fusion concerns are discussed and topics that need to be addressed in more detail are described.

DATA ASSOCIATION AND FUSION IN AN ASW ENVIRONMENT

Kenneth McPhillips and Robert Fagan

Raytheon Submarine Signal Division
1847 West Main Rd.
Portsmouth, Rhode Island 02871-1087

INTRODUCTION

Soviet submarine quieting represents the single most serious threat to U.S. Navy superiority since World War II and the fleet has been stating for years that no single platform or Anti-Submarine Warfare (ASW) tool is capable of effectively handling this threat. The quiet threat mandates improved sensors to enhance detection capabilities and will require an integration of forces for rapid targeting. These improved sensors as well as this greater reliance on off-board sensors and cuing data will inevitably result in a greater number of contacts which in turn will present data overload and operability problems in ASW Combat Control Systems. Data association, the ability to determine whether multiple contacts are from the same target and data fusion, the ability to combine multiple associated contact solutions will be required to present a single concise tactical picture to all friendly forces in multiple target environments. Association and fusion can reduce ambiguities and increase confidence in the tactical picture, can enhance spatial resolution utilizing the large apparent aperture between cooperating vessels, can enhance the spatial and temporal coverage of a single vessel and can enhance detection and tracking capabilities through feedback information.

The ASW problem can be difficult due to the poor quality of sonar data as well as the sparsity of the data as contacts are lost and reacquired in convergence zones. Thus, poor observability, often characteristic of ASW problems can make data association difficult, and the quiet threat will further compound association difficulties. This paper will address the different types of data association, the levels of data association, the errors which can occur in association, and manual association.

DATA ASSOCIATION PRELIMINARIES

The first level of any data association routine should weed out contacts which are obviously from different targets based on their classification and attributes as well as their geographic position. Obviously, it is unadvisable to try to associate a surface ship with a submarine if one is confident about each contact's classification. Likewise, it is ridiculous to try to associate two contacts whose geographic positions are significantly different, provided the confidence in the position estimates is high. Another way of prefiltering association candidates is based on a sensors known probability of detection and is discussed in Ref 1. Simply stated, it is foolish not to associate two similar contact solutions generated from sensors with high probabilities of detection, since the probability that a second target was present and undetected is low. Therefore, the contacts most probably are from the same target and the estimates can be fused immediately. Thus, an association test should be carried out only when the source of the two contacts is doubtful. Although these preliminary data association rules are heuristic, they could significantly reduce the number of viable association candidates, could prevent erroneous data fusions and could prevent missing valid data fusions. (See Ref 2 for more thoughts on data association preprocessing.) The large number of anticipated contacts, given new sensor technology, will require prefiltering the association candidates in order to avoid more intensive association processing later.

The next level of data association should ensure that the state estimates and covariance matrices were generated using independent data sets. Performing a data association test on the outputs of two algorithms which have used common data will most likely result in an overly optimistic composite estimate. The status of contact

maneuvers should be checked to ensure that no maneuvers have occurred. A contact maneuver will inevitably produce erroneous correlations between the two sensor estimates and will invalidate the data association test. Most ASW Target Motion Analysis algorithms assume a constant speed and course target motion model and rely on some form of residual test to detect target maneuvers. Unfortunately, poor scenario geometry and poor measurement quality can make timely detection of target maneuvers difficult and this will have serious repercussions in data association. The next check should ensure that the contact solutions and corresponding covariance matrices are reasonable and that the algorithms computing the solutions are performing as expected. The measurement residuals, $z-h(\hat{x})$, can be used as a performance indicator as suggested in Ref. 3. One quick check which can be made to ensure algorithm convergence is to compute the objective function

$$q(x) = (z-\hat{z})^T R^{-1} (z-\hat{z}) \quad \hat{z} = h(\hat{x}) \quad (1)$$

where z denotes the measurement vector, \hat{z} denotes the measurements which would be observed at the estimated state, \hat{x} , and R denotes the covariance of the measurements

$$R = \begin{bmatrix} \sigma_{z_1}^2 & & \\ & \ddots & \\ & & \sigma_{z_m}^2 \end{bmatrix} \quad (2)$$

and compare the computed value of the objective function to its expected value. The objective function can be shown to have a χ^2 distribution with $m-n$ degrees of freedom where m is the number of measurements and n is the number of states. Its expected value can be computed:

$$E[q(x)] = E\left\{(z-\hat{z})^T R^{-1} (z-\hat{z})\right\} = m-n \quad (3)$$

A quick check of algorithm performance would be to compute the objective function and compare the computed value to the number of measurements minus the number of states. If the objective function is significantly larger than its expected value, then the algorithm is not performing as expected and its state estimate and corresponding covariance matrix should be ignored by the data association routine. This quick check has been done for both a batch and recursive processor and has proved to be a reliable performance indicator. Many types of performance indicators have been proposed, but the simple test previously described performs well and is not CPU intensive. Generated target solutions should always be examined before use in any kind of association test, since poor algorithm performance can result in some TMA scenarios.

GEOMETRIC FUSION

Once all of the initial checks on the association candidates have been done and the coordinate systems have been aligned to ensure gridlocking (see Ref 4), a simple statistical test such as a χ^2 test, can determine whether or not two contacts are from the same source (see Ref 5). The test used in this report takes two independent state

estimates \hat{x}_i, \hat{x}_j which apply at their respective times t_i, t_j and the corresponding covariance matrices P_i, P_j and extrapolates the oldest state vector and covariance matrix forward in time to the most recent time via the equations,

$$\hat{x}_i = \Phi \hat{x}_j, \quad P_i = \Phi P_j \Phi^T \quad \text{if } t_i < t_j$$

(4)

or

$$\hat{x}_j = \Phi \hat{x}_i, \quad P_j = \Phi P_i \Phi^T \quad \text{if } t_j < t_i$$

where

$$\Phi = \begin{bmatrix} 1 & 0 & |t_j-t_i| & 0 \\ 0 & 1 & 0 & |t_j-t_i| \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

The χ^2 statistic is then computed as

$$\chi^2 = (\hat{x}_i - \hat{x}_j)^T (P_i + P_j)^{-1} (\hat{x}_i - \hat{x}_j) \quad (6)$$

and compared to a threshold δ . The threshold δ is chosen from a χ^2 table such that the probability that $\chi^2 > \delta$ given the assumption that $\hat{x}_i = \hat{x}_j$ (contacts from the same target) is some chosen probability. Thus the probability of failing to associate two contacts which are actually from the same target can be set a-priori. If $\chi^2 < \delta$ then the two solutions can be fused via the equation

$$P_c = [P_i^{-1} + P_j^{-1}]^{-1} \quad (7)$$

$$\hat{x}_c = P_c [P_i^{-1} \hat{x}_i + P_j^{-1} \hat{x}_j]$$

where \hat{x}_c and P_c are the combined solution and associated covariance matrix. If $\chi^2 > \delta$ then the solutions should not be fused and the data streams should continue to be treated as separate targets. In the algorithm implemented in this paper, all matrix inverses are computed using Cholesky decomposition via the equation

$$A = L U \quad (8)$$

where L is lower triangular, $U = L^T$ is upper triangular,

$$A^{-1} = U^{-1} L^{-1} \quad (9)$$

and U^{-1} is computed by back substitution.

This test assumes that complete, independent, cartesian coordinate solutions (x, y position and velocity) are available, and can be used to associate and fuse only two solutions. However, this simple algorithm can be modified to compute the most likely association candidates and could be extended so that it could fuse multiple solution estimates.

Although the χ^2 test assumes complete Cartesian Coordinate state estimates and covariance matrices, the test can be used to associate and fuse outputs from various algorithms. The TMA algorithms considered in this report are the Non-Linear Least Squares algorithm (NLLS), the Modified Polar KAST algorithm (MPKAST) and the Modified Spherical KAST algorithm (MSKAST). The NLLS algorithm discussed in this report is a multisensor Gauss-Newton type batch processor which employs Householder Orthogonal Triangularization techniques and was developed as part of Raytheon's Advanced ASW Combat System (see Ref 6). MPKAST

is an extended Kalman filter which processes bearings and range measurements and is currently the primary automatic TMA algorithm in CCS MK 1. MSKAST is also an extended Kalman filter developed by NUSC Newport to process Wide Aperture Array data. Since the NLLS algorithm is formulated in Cartesian coordinates and since both KAST algorithms convert their polar state estimates and covariance matrices into their Cartesian counterparts via appropriate transformations, it is simple to fuse data from these algorithms. Cartesian coordinates were chosen for the data association test since the extrapolation of the state vector and covariance matrix is simple and faster than the extrapolation in modified polar or modified spherical coordinates. However, a little amount of shuffling of the state estimates and covariance matrices is needed to fuse estimates from these three algorithms since the state vectors are of the form

$$\begin{aligned} \text{NLLS} & \quad (x \ y \ \dot{x} \ \dot{y} \ z) \\ \text{MPKAST} & \quad (\dot{x} \ \dot{y} \ x \ y) \\ \text{MSKAST} & \quad (x \ y \ z \ \dot{x} \ \dot{y}) \end{aligned}$$

It is also possible to use the association/fusion algorithms described in this report in cases where the number of state variables in the association candidates differs. For instance, if MPKAST is providing estimates of target x, y position and x, y velocity (calculated from the modified polar coordinates) and MSKAST is estimating target x, y position, target x, y velocity and target depth, then the depth estimate from MSKAST can simply be ignored. If depth and/or doppler parameters are available from both candidate algorithms, the association routine can be modified slightly to include this additional information. To handle these cases, the number of degrees of freedom in the χ^2 test must be set to the minimum number of state variables in the candidate state vectors, and the appropriate threshold must be used. In essence, a great deal of flexibility exists in combining solutions from different algorithms processing different sensor data.

In addition, it is possible to modify the test somewhat to handle localization (position only) outputs. If only target x, y positions are estimated, then a χ^2 test with only 2 degrees of freedom can be used. However, it is necessary for the localization algorithms to provide the covariance matrices associated with the position estimates, which are not provided in any current combat system. The towed array crossfix software residing in CCS MK1 been enhanced by Raytheon so that localization ellipses can be generated along with the position estimates for any combination of azimuthal bearings and conical angles from any sensors (Ref 7), and this software can be utilized to try data association on position-only outputs. If a history of contact positions are available, then all of the positions can be used in the χ^2 test (as is done in Ref 1), which will result in a more reliable test.

In some cases it may be useful to attempt a data association in modified polar coordinates. If the ownership is restricted to a single leg and is tracking a spherical array contact via MPKAST and is simultaneously tracking a towed array contact via MSKAST, then a complete TMA solution is unobservable, since range is unobservable in a bearings only or conical angles only case. However, since range is lacking and there is nothing to lose, and if an operator feels he has the same target on both sensors, it is possible to take the observable quantities from MPKAST (i.e., bearing, bearing rate and normalized range rate along with the corresponding covariance sub-matrix) and use the bearing, bearing rate and normalized range rate estimates

along with the corresponding modified polar covariance matrix from MSKAST (which it provides) and attempt a data association on the three observable quantities. If the test indicates a probable correlation between the contacts, then the data sets can be combined to yield target range. This combination can take place at either the measurement level (via NLLS) or at the "track" level. In a first leg situation like this, it may be best to perform the data fusion at the measurement level using NLLS with the numerically stable Householder technique. Combining at the track level may result in erroneous solutions due to round-off errors. Once the data fusion has been performed, the composite solution can be evaluated using the composite objective function as a performance indicator as previously discussed.

GEOMETRIC DATA FUSION LEVELS

There are basically two levels at which multi-sensor or multi-platform data can be fused once two contacts have been associated. For the purpose of this discussion, it will be assumed that a spherical array contact and towed array contact have been correlated via some association test. From that point in time on, it is possible to: 1) continue to treat the contacts separately and fuse data at the track level, after generating state vectors and covariance matrices for both the sphere and towed array contacts (see Fig. 1) or 2) fuse data at the measurement level and generate the single combined state vector and covariance matrix via the multi-sensor NLLS algorithm. Although fusing data at the different levels should produce identical results assuming no round-off errors, data fusion done at the track level is preferred for the following reasons:

1. this approach provides the operator with information on the constituent solutions (3 localization solutions vice only 1)
2. decorrelation is trivial (already done since constituent solutions are saved)
3. allows for future statistical tests to affirm/deny associations.
4. allows for error budget so that sensors which are contributing little to a solution can be reassigned.

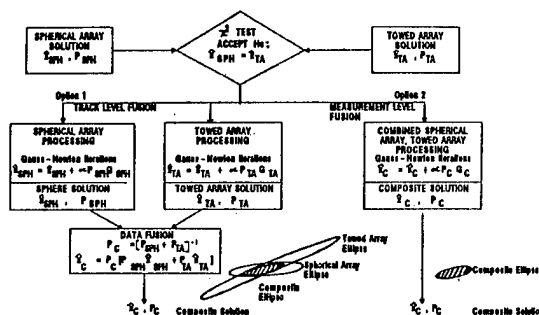


Figure 1. Track/Measurement Level Fusion

The disadvantages of track level data fusion are:

1. requires more CPU time
2. requires more storage (about three times the storage since both solutions along with the combined solution is saved)
3. less numerical stability than measurement level fusion

The main disadvantage of measurement level fusion is that measurements are usually not broadcast between cooperating vessels. Usually, only solution data is transferred making measurement level fusion impossible.

In general, it is difficult to say how much slower track level data fusion is because track level fusion via a batch processor may require many iterations to generate the individual solutions but typically only a few iterations are required to generate the combined solution (measurement level fusion) since the multi-sensor/multi-platform problems are more observable and easier to solve. The track-level data fusion routine presented in Eqns. 4-7, takes 1/10 CPU second assuming two 4-dimensional state vectors when done in double precision on a VAX 11-780.

In order to demonstrate the equivalence of data fusion performed at the track level with data fusion performed at the measurement level, the NLLS algorithm was run in both configurations on the scenario depicted in Figure 2. In the first run, the NLLS algorithm was executed using ownship bearings from target 1 and the state estimate and covariance matrix at the end of the scenario were generated. In run two, the NLLS algorithm was executed using only consort bearings from target 1 and the state estimate and covariance matrix at the end of this run were also generated. These two state estimates and two covariance matrices were then associated and fused via the algorithm previously discussed. Then, the NLLS algorithm was executed using both ownship and consort bearings from target 1 and the results obtained from this run are compared to the results obtained by fusing at the track level in Table 1.

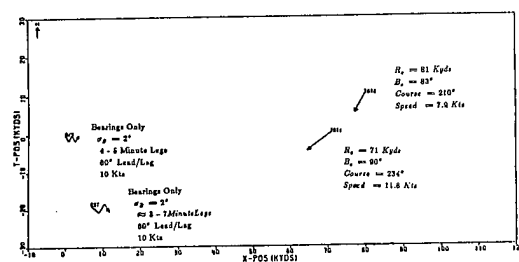


Figure 2. Multi-platform Scenario Geometry

	Range Estimate yds	Range St. Dev. yds	Bearing Estimate deg	Bearing St. Dev. deg	Course Estimate deg	Course St. Dev. deg	Speed Estimate kts	Speed St. Dev. kts
Fusion at Track Level	81,138	511	94.064	.138	235.322	3.191	11.861	1.363
Fusion at Meas Level	81,150	538	94.081	.128	235.449	3.484	11.836	1.378
[Difference]	12	27	.003	.008	.072	.293	.025	.015

Table 1. Track/Measurement Level Fusion
Multi- platform Case

All of the NLLS runs presented in this paper have been made in double precision and employ Householder Orthogonal Triangularization.

Obviously, the state estimates and estimated solution quality are equivalent except for small roundoff errors introduced mainly in the matrix inversion routines. Data fusion performed at the measurement level will result in less roundoff errors than data fusion performed at the track level since the combined information matrix is apt to be less singular and thus more easily invertible than either of the constituent information matrices. This is because the TMA problem is more observable and easier to solve when multiple data sets are used simultaneously. In very poorly observable scenarios, data fusion done at the track level may suffer heavily from roundoff errors and the composite estimate and covariance matrix may be biased. Therefore for long range low bearing rate scenarios, it may only be feasible to fuse data at the measurement level using the numerically stable Householder Orthogonal Triangularization approach, since the individual solutions may not be computable if the computer word length is small.

In order to demonstrate the numerical superiority of data fusion performed at the measurement level as opposed to data fusion performed at the track level, the NLLS algorithm was run using spherical array bearings and towed array conical angles on the poorly observable scenario described in Figure 3. The NLLS algorithm was run in both a bearings only mode and a conical angles only mode and the two solutions and covariance matrices were fused at the track level. The NLLS algorithm was also run in a spherical array/towed array multi-sensor mode (data fusion at the measurement level) and the results are compared to the track level fusion in Table 2.

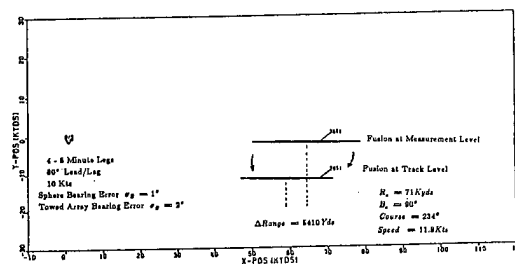


Figure 3. Single Platform Scenario Geometry

	Range Estimate yds	Range St. Dev. yds	Bearing Estimate deg	Bearing St. Dev. deg	Course Estimate deg	Course St. Dev. deg	Speed Estimate kts	Speed St. Dev. kts
Fusion at Track Level	62,919	6,085	91.824	.120	102.719	22.773	27.584	49.045
Fusion at Measurement Level	57,609	5,713	91.933	.110	140.353	219.103	7.574	24.102
[Difference]	5410	952	.109	.010	37.634	196.330	19.99	24.841

Table 2. Track/Measurement Level Fusion
Single Platform Case

(The differences in the 90% localization ellipses are depicted in Fig. 3). Obviously, the fused solutions differ appreciably, especially course and speed and this difference can be traced to the large roundoff errors which occur when inverting poorly observable matrices. When data fusion is performed at the track level, two sets of roundoff errors occur. The first set of roundoff errors occurs when the covariance matrices for both the sphere

and towed array are inverted. Then the track level fusion routine inverts the sum of the information matrices adding on even more roundoff errors. Obviously, these errors can be significant even when using double precision arithmetic.

DATA ASSOCIATION ERROR TYPES

There are two types of errors which can occur in data association. Type I errors are made when the null hypothesis (i.e., the contacts are from the same target) is incorrectly rejected or stated differently, when the algorithm fails to combine contacts which are actually from the same target. The consequences of Type I errors are that two poor solutions from supposedly two different targets are maintained when one potentially good solution is possible. Type II errors occur when the null hypothesis is incorrectly accepted, or, when the algorithm combines two contacts which are from different targets. The consequences of Type II errors are that a poor solution along with an overly optimistic solution quality can be generated. Type II errors are more serious than Type I errors since they may indicate a false position with a tight error bound.

In order to demonstrate the consequences of both types of error, the TMA scenario depicted in Fig. 2 was simulated. Ownship, "O/S", is processing spherical array bearings with a bearing standard deviation of 2° into an idealistic 20 second block averager and has made four 5 minute legs with 60° lead/lags at a speed of 10 knots. The consort, "CST", is also processing spherical array bearings with a bearing standard deviation of 2° into the same block averager and has made roughly 3 seven minute legs and is also doing 60° lead/lags at 12 knots. Figure 4 shows the 90% localization ellipses on target 1 as seen by the ownship and consort, along with the fused solution. If the algorithm failed to associate the two contacts, then both poor solutions would be maintained on the "two" targets rather than have the one high quality solution. Type I errors will result in the loss of potentially high quality solutions which could lead to missed or unnecessary weapons launches and will necessitate storing and processing redundant data sets.

Figure 5 graphically depicts the gravity of Type II errors. In this scenario, ownship has processed the bearing measurements from target 1 while the consort has processed bearing measurements from target 2. The localization ellipses generated by both ownship and consort are depicted along with the fused solution. Obviously, the fused solution is off significantly and the estimated quality of the fused solution is excellent. Therefore, this type of error could lead to a useless weapons launch at a false target position where the probability of hitting either one of the targets would be slim. False associations will lead to very misleading tactical pictures.

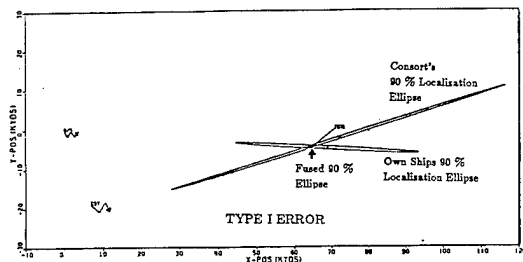


Figure 4. Missed Association

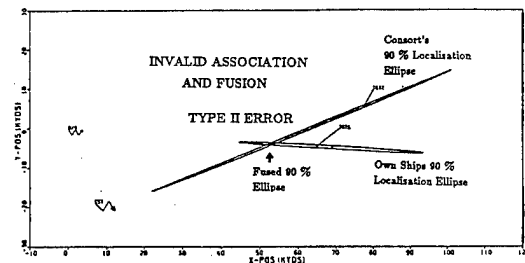


Figure 5. Invalid Association

It is wise to avoid both types of error, however, any association algorithm will exhibit both types of errors to some degree and a good data association algorithm should provide the tools to rapidly associate two contacts or gracefully de-associate one supposed target into its constituent contacts, since data association errors are inevitable. When the χ^2 test is used, it is possible to specify the probability of the Type I errors by setting the threshold used in the test, however, it is impossible to directly specify the probability of the more serious Type II errors because the conditional probability that χ^2 is greater or less than a threshold given that the two contacts are not from the same target can not be calculated, unlike the classical signal detection problem. Unfortunately, setting the χ^2 threshold sets the probability of both types of error, and the probability of Type II errors will be a function of the scenario. Therefore, the attribute comparisons between contacts will play a key role in minimizing both types of error and its importance can not be underestimated. In many ASW applications, good classification data on surface vessels can be accompanied by poor positional data and the classification data must be fully exploited.

The threshold used in the association test may be dictated by both the number of contacts and the tactical circumstances and rather than propose a threshold for the χ^2 test, the relationship of the threshold to both types of error will be examined.

EXPERIMENTAL DATA ASSOCIATION RESULTS

In order to examine the performance of the data association test, 400 simulations of the scenario depicted in Fig. 2 were run where both the ownship and consort were tracking target 1. In each simulation, the contacts are treated separately and data association and fusion was performed at the track level. Various probabilities of missed associations were chosen and the results are presented in the following table.

Probability of Missed Associations	χ^2 Threshold	Actual % of Correctly Associated Runs	Actual % of Missed Associations
.50	3.375	50%	50%
.20	5.989	74%	26%
.10	7.779	81%	19%
.05	9.488	87%	13%
.01	13.277	91%	9%

Table 3. Multi-platform Fusion Results Missed Associations

A perfect data association routine would associate all of the solution pairs, and if the χ^2 threshold was set to infinity, the algorithm would indeed associate all solution pairs. Unfortunately the algorithm will associate all solution pairs whether or not they are from the same target. Therefore, as in any kind of association test, the chosen threshold is a compromise based on the consequences of both types of errors. As can be seen from the table, the number of correctly associated runs is low for high probabilities of missed associations. An unusually large number of high χ^2 values were found and 38 runs exceeded the threshold of 13.377 when only 4 runs over this threshold would be expected. Four runs produced χ^2 statistics of over 100, which is extremely improbable. Although the reason for both the low number of correct associations and large number of high χ^2 values is not presently known, it is possible that the non-Gaussian nature of the target solutions is to blame. In non-linear applications, the distributions of the estimated state variables are not normal, which the χ^2 test assumes. This problem is currently being investigated.

In order to determine the frequency of type II errors (bad associations), 400 simulations of the same scenario (see Fig. 2) were run where the ownship is tracking target 1 and the consort is tracking target 2. The same probabilities of type I error (missed associations) as in the previous table were used and the results are presented in the following table:

Probability of Missed Associations	χ^2 Threshold	Actual % of False Associations
.50	3.375	4%
.20	5.989	36%
.10	7.779	66%
.05	9.488	83%
.01	13.277	94%

Table 4. Multi-platform Fusion Results
Incorrect Associations

As anticipated, lowering the probability of missed associations results in an increase in the probability of false associations, since it is impossible to decrease the probabilities of both types of error simultaneously. In this scenario, if the probability of missing an association is chosen to be .10, then given the track of the second target "TGT2", there is a 66% probability that the two targets will be incorrectly fused, resulting in a poor solution and tight error bound (see Fig. 5). If a different target 2 track was specified, then the number of false associations might be reduced significantly, which demonstrates the point that the probabilities of missed associations can be set; but, the probabilities of false associations are scenario dependent and can only be determined experimentally. If the second target track was chosen so that it was not even close to the first target or if the solution quality was high, then very few false associations would occur. When two targets are close to each other and traveling in the same general direction, and when the solution quality of both targets is poor, then the probability of false associations will be high. Therefore, even specifying a probability of a missed association is difficult.

In order to show how drastically the TMA scenario parameters affect data association, the scenario described in Fig. 2 was repeated, however, the measurement noise was reduced. Reducing the measurement noise substantially increases the quality of the solutions

generated by both the ownship and its consort (see Figure 6) and it is apparent that the two targets are easily distinguishable since their 90% localization ellipses are disjoint. In general, making better measurements or collecting more data will substantially aid in association testing and it is important to keep testing to affirm or deny any previous associations, especially when measurement data is poor and the associations are weak. To demonstrate this, data association was tried on 400 runs for the case where both friendly vehicles are tracking target 1 using various probabilities of missed associations. When both friendly vehicles are tracking target 1, the data association test performs as expected as shown in the following table:

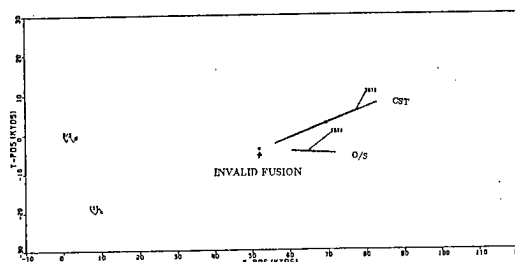


Figure 6. Multi-platform Scenario Geometry
High SNR Case

Probability of Missed Associations	χ^2 Threshold	Actual % of Correctly Associated Runs	Actual % of False Associations
.50	3.375	53%	0%
.20	5.989	83%	0%
.10	7.779	91%	0%
.05	9.488	95%	0%
.01	13.277	99%	0%

Table 5. Multi-platform Fusion Results
High SNR Case

Using the same probabilities of missed associations, the data association routine did not falsely associate any solution pairs for the case where ownship was tracking target 1 and the consort was tracking target 2. In the previous example (larger measurement noise) there were a great deal of false associations since the solution quality was poor. In general, poor measurement quality and poor scenario geometry will result in both poor TMA solutions and poor data association results.

MANUAL DATA ASSOCIATION

It is interesting to see what happens to the bearing residuals when contacts are correctly and incorrectly fused, since a manual association decision may be based on the current dot stacking technique used in today's Combat Control systems. Figure 7 shows the bearing residuals as seen by both the ownship and consort for the scenario depicted in Fig. 2. The leftmost plot shows the bearing residuals produced by taking the difference of the ownship's measured bearings and the predicted bearings based on the ownship solution and the bearing residuals produced by taking the difference of the consort's measured bearings and the predicted bearings based on

the consort's solution. The middle plot shows the bearing residuals produced by taking the difference of both the ownship's and consort's measured bearings and the predicted bearings based on the combined solution for the case where the valid association is made (see Fig. 4). It is interesting to note that although the solution improves significantly when the contact solutions are validly fused, the objective function increases slightly for both ownship and consort (\hat{Q}_{os} goes from 69.42 to 69.62 while \hat{Q}_{cs} goes from 44.27 to 45.41). Although this seems contradictory, it is easily explainable. For the no noise case, the minimum of the objective functions will be zero and will occur at the true target state (see Figure 8). When noise is added, ownship will minimize its objective function and obtain its minimum value \hat{Q}_{os} at its best estimate \hat{x}_{os} . Similarly, the consort will minimize its objective function and obtain its minimum value \hat{Q}_{cs} at its best estimate \hat{x}_{cs} . When the states are fused, and the objective functions are computed using the fused state estimate (which is better than either single estimate), the objective functions \hat{Q}_{os} and \hat{Q}_{cs} must increase since their minimum values occur at \hat{x}_{os} and \hat{x}_{cs} respectively. The rightmost plot of Figure 7 shows the bearing residuals produced by taking the difference of the ownship and consort measured bearings and the predicted bearings based on the combined solution for the case when an invalid association is made. In this case, the combined solution is significantly biased and the indicated quality is high (see Fig. 5). Both objective functions increase as expected although the objective function for ownship increases only slightly (from 69.42 to 73.32). As expected, the value of χ^2 is low (1.0943) when the valid association is made, and χ^2 is high (8.5874) when an invalid association is made.

From looking at the residual plots, it is easy to see how difficult it is for an operator to confidently make an association decision, since all the "dot stacks" look quite similar. The bearing residuals will not always get significantly larger when an invalid association is made, since an algorithm will fit a viable target track through both sets of contact bearings. The true value of the operator will be in evaluating the association based on classification and attribute data as well as experience.

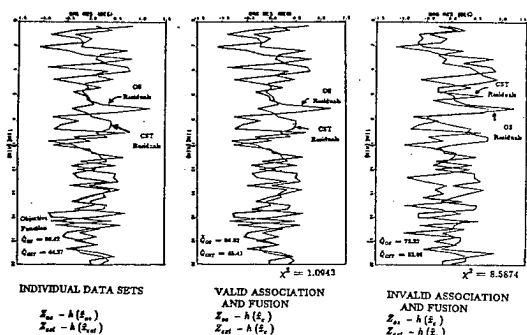


Figure 7. Manual Association Residual Plots

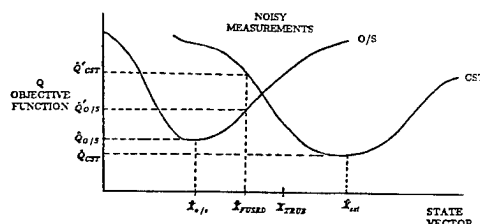


Figure 8. Objective Function

CORRELATION OF TOWED ARRAY CONICAL ANGLE HISTORIES WITH RADAR POSITION HISTORIES

Typically, surface ships maintain a constant course and speed track for long intervals while making passive measurements of conical angles on many submarine or surface contacts. Without an ownship maneuver, a position track cannot be derived from only conical angles. Even if an ownship maneuver is performed, the position of many contacts may only be weakly observable (due to poor geometry) and therefore not practically computable. If radar position tracks of surface contacts are measured either by ownship radars or radars on other platforms, it would seem tactically important to attempt correlation of the towed array conical angle histories with the radar position histories. Successful correlation would allow culling out of the surface traffic from ownship conical angle contact files.

The correlation decision may be performed by first converting the radar position $[x(t), y(t)]$ to conical angles, $\beta(t)$ as seen from ownship and then a hypothesis test can be performed on the differences between these conical angle histories as derived from radar and the conical angle histories as measured by ownship. To do the x, y conversion to conical angle, the Inverse Raytrace algorithm described in Ref 8 could be utilized. The correlation decision steps are as follows:

- Interpolate x, y positions to times of ownship conical angle positions.
- Convert x, y positions to range, R , and bearing, B , from ownship.
- Find D/E from R using Inverse Raytrace Program
- Compute conical angle β from $\cos \beta = \cos B \cos D/E$
- Propagate confidence ellipses (P matrices) of x, y positions into conical angle variances.
- Compute weighted differences between ownship conical angles and the conical angles as derived above from radar position reports. The weights are derived from the variances of the ownship and the radar conical angles.
- Compare weighted differences to a predetermined threshold and declare correlated those tracks below the threshold.

TRACK SEGMENT ASSOCIATION

R. Mucci in Ref 9 describes an algorithm for deciding whether two sensors reporting track segments are observing the same target. The hypothesis test is "Is there a time for which the two estimated trajectories have a common position?" The test is equivalent to a χ^2 test on the weighted difference of the extrapolated track segments to the common time; then the track segments can be re-estimated using the combined set of measurements. The technique allows for target maneuvers. Figure 9 shows two track segments for a maneuvering target as reported by two sensors and the extrapolated common time error ellipses. Figure 10 shows the error ellipse of the combined measurements sets for the common point of intersection. This then is a version of the Temporal Data Association problem described in Ref 10 by V. Gabriel.

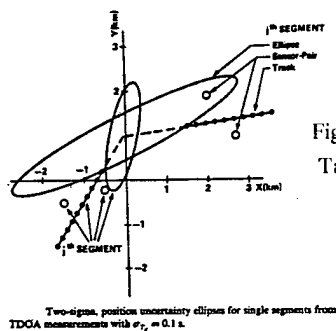


Figure 9. Maneuvering Target Track Segments

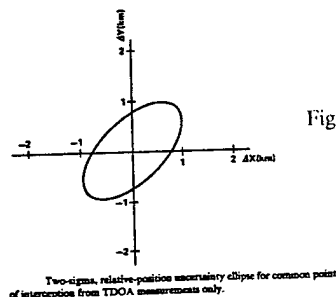


Figure 10. Error Ellipse for Combined Measurement Sets

ELIMINATING FALSE TARGETS FROM MULTI-SENSOR BEARING DATA

A fairly new technique called simulated annealing (described in Ref 11) can be used to find the lowest minimum of a multi-minima function and should be investigated for applicability to contact management. The application to TMA is illustrated in Figure 11 which shows the multi-minima cost function obtained by estimating two target locations as measured by bearing lines from three sensors. The false target locations have a higher minimum than the true target locations; and a conventional NLLS gradient descent algorithm could be trapped at these false locations.

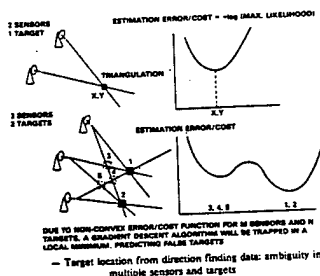


Figure 11. Multi-minima Cost Functions

STATE ESTIMATE HISTORIES

Although the association algorithm presented here uses only two single full state estimates, the χ^2 test can be modified to handle position-only state estimate histories as demonstrated in Reference 1. Likewise, full state histories can be used to test whether 2 contacts are from the same source. Using histories of uncorrelated position estimates will result in an improved association test, since more data will yield more confident associations. However, if state estimates are generated sequentially by a TMA algorithm, like NLLS, MPKAST or MSKAST, then

correlations between successive solutions will exist since over-lapping measurement sets are used to generate the solutions. Therefore, since these estimates are correlated, using a history of correlated estimates vice a single uncorrelated estimate pair may not yield significantly better association results. However, if an algorithm uses a finite data window (which NLLS is capable of doing) and if the state estimates are chosen such that there is virtually no overlap between the measurement sets used in generating the solutions, then the solution history approach might yield significantly better data association results since there would be virtually no correlation in the state estimates. Likewise, a short history of widely time separated state estimates could yield better association results.

CONCLUSIONS

The increased number of contacts resulting from sensor improvements and the use of off-board data will require reliable automatic multi-sensor, multi-target algorithms. Solutions generated by these algorithms must be automatically screened using a performance index. Also reliable target maneuver detection is necessary to avoid incorrect association which can lead to an extremely misleading tactical picture. Manual performance assessment and maneuver detection is slow and should be avoided whenever possible. Likewise, manual geometric association is difficult using the "dot stack" approach and its utility is questionable. Simple statistical tests, such as the central χ^2 test, are useful for geometric association, however, these tests must be accompanied by reliable attribute measurement schemes, and the association results should be periodically checked as more data is received, in order to avoid false associations. Since the probability of false associations can not be directly specified and varies greatly with both measurement quality and scenario geometry, and because the TMA problem can be quite troublesome, false associations will inevitably occur.

Geometric data fusion can be accomplished at both a measurement level and a track level. Track level fusion is preferred because it provides more information, simplifies decorrelation, provides for subsequent statistical tests, facilitates the use of an error budget, and finally is natural when processing measurements from different sensors. In poorly observable scenarios, track level fusion can suffer numerical precision problems and the resulting biased composite solutions have been demonstrated even though numerically stable orthogonalization techniques were employed and double precision arithmetic was used. Under adverse conditions, measurement level geometric fusion may be the only practical means of computing reliable composite solutions.

REFERENCES

1. Howard L. Weiner, "Track to Track Association in Ocean Surveillance", Naval Research Lab., Report No. NRL 7989, May 12, 1976.
2. Paul L. Bongiovanni, "Program Description of a Contact Management Function; Localization and Correlation," Raytheon Report TIC-UR87-0026, June 3, 1986.
3. Vincent J. Aidala et.al, "The Utilization of Data Measurement Residuals for Adaptive Kalman Filtering," NUSC Report TR4502 February 8, 1974.
4. James T. Miller and Eugene David, "The Role of Navigation in the Precision Registration of Sensor Data at Sea," Journal of the Institute of Navigation, Vol 30 #2, 1983.
5. Samuel S. Blackman, "Multiple Target Tracking with Radar Applications," Artech House, Dedham, MA, 1986.
6. Paul L. Bongiovanni and Ken McPhillips, "Technical Report to Document Contact Management IR&D Work," Raytheon Report TIC-UR87-0024, Rev 1, June 30, 1987.
7. "Localization Ellipses for Crossfix Positions," Raytheon memo KJM:87/04, March 25, 1987.
8. "Towed Array Sound Path Correction Implementation," Raytheon memo KJM:83/04, May 20, 1983.
9. R. Mucci, J. Arnold, Y. Bar-Shalom, "Track Segment Association with a Distributed Field of Sensors," Journal Acoustical Society of America, Oct. 1985, pp 1317-1324.
10. V. T. Gabriel, R. D. Berlin, Y. Bar-Shalom, "Data Association in an Ocean Environment," General Electric Co., Syracuse, N.Y. 1981.
11. Harold H. Szu, "Non-Convex Optimization," Naval Research Laboratory, Code 5709, Washington, D.C.

BIOGRAPHIES

Kenneth McPhillips was born on November 1, 1957. He received a Bachelor of Arts degree from Providence College in 1979. He has performed systems engineering work on various Fire Control and Combat Control systems. His current work at Raytheon is in developing target tracking algorithms as well as data association and fusion algorithms for an advanced Anti-Submarine Warfare Combat system.

Robert Fagan obtained a Bachelor of Science degree in Electrical Engineering from Syracuse University in 1951 and is a 1958 graduate of the three year General Electric ABC Advanced Engineering Program. He has performed systems engineering for both radar and sonar systems. His current work at Raytheon Company is in target track estimation and data association and data fusion algorithms using passive sonar measurements.

A Multisensor Approach to Aircraft Detection
and Identification

G. A. Roberts and L. H. Bradford

Ford Aerospace
Ford Road
Newport Beach, CA. 92658

A multisensor approach to detection and identification of aircraft is described. The sensors that are utilized are a dual mode long wavelength infrared (IR) sensor and a imaging visual sensor. The IR sensor will be able to perform in an imaging and track while scan mode. The detection is accomplished with the IR sensor in a track-while-scan mode (TWS). The detection algorithm is based on a least-mean-square (LMS) filter with a post screening algorithm. Identification is performed using the visual or imaging IR sensor. The aircraft identification algorithm is a knowledge based approach based on the US Army's "Visual Aircraft Recognition Manual." The algorithms and system design will be described however the emphasis will be on the detection and identification algorithms.

A Multisensor Approach to Aircraft Detection and Identification

G. A. Roberts and L. H. Bradford

Ford Aerospace
Ford Road
Newport Beach, CA. 92658

INTRODUCTION

Current aircraft detection and classification systems operate with a high degree of autonomy. The detection system will typically just provide the target position to the classifier. A synergistic system is presented which consists of a dual mode IR sensor (IRST and IIR), and a visible imager in a shared aperture and dual stabilized platform. The detection is accomplished using a LW-IRST (long wavelength infrared search and track) sensor with a detection processor. A hybrid detection algorithm is used with a automatic sensitivity control. The classification is done using imagery from the visual imager (long range day use) or the IR imager (shorter range night use). The classification algorithm is knowledge based which utilizes a aircraft parts analysis technique derived from the US Army's visual aircraft recognition field manual. The knowledge based system stores three dimensional representation of aircrafts for matching. The classifier relies on the detection processor to provide target range, velocity, and position. The knowledge based system utilizes this information to calculate aspect for the purposes of developing silhouette models from the 3-D aircraft representations, and to size the models. The use of aspect and range greatly increases the classification performance. The classifier is also able to provide the TWS processor with target/clutter information which will enable it to adjust the automatic detection threshold parameters to optimize detection to false alarm performance.

The synergism between the detection and classification greatly improves the performance of both functions. The knowledge based classifier is greatly assisted by being provided the range and velocity of the aircraft. This enables the classifier to determine

target aspect and size more reliably. Without this information the classification performance will be severely limited since the classification will need to treat the aspect as a degree of freedom. Additionally the classifier will be able inform the detection processor that it is detecting clutter. This will allow the detection processor to adjust its detection sensitivity so that it does not report an excessive amount of false detections. The basic system will first be described, then the detection and classification algorithms will be presented.

SYSTEM DESCRIPTION

The system consists of a dual mode IR, visual sensor, detection/identification processor, and a controller. The basic system is shown in figure 1. The IR sensor will have the capability of scanning a narrow and wide field of view in the imaging mode. In the TWS mode it will function as an infrared search and track sensor and execute a gimbal scan while performing a rapid parallel scan. The detection/identification processor will perform target detection using the IR seeker and assemble a track list of the detections. Target range will be estimated using a passive ranging technique. When a detected target is in recognition range the sensor will stare at the target and a visual sensor will be used for identification. At night the IR sensor can be used in an imaging mode but the recognition range will not be as long as a visual sensor in clear sky. The difference is due to the greater resolution possible at the visual wavelengths as compared to the infrared wavelengths. In order to achieve the increase in resolution a higher degree of stabilization must be achieved for the visual sensor.

The detection/identification processor will consist of a dual buss architecture with the repetitive high computational operations such as edge detection and LMS filtering being performed by parallel processors off a high speed image buss. The lower computational and more hierarchical operations will be executed by using programmable processor modules off a VME buss. The demonstration processor utilizes Datacube processor cards in addition to specially built processors off a high speed image buss. Motorola 68020 processors are also used for the lower computational computations. This configuration allows flexibility of programing the higher level operations while having expandable lower level primitives.

AIRCRAFT DETECTION

Aircraft detection will be done by using the imagery from the IR sensor while it is in the TWS mode. The detection algorithm structure is shown in figure 2. The algorithm uses a LMS filter as a preliminary screening algorithm [1,2] to identify areas of the image that may contain targets. The filter responses are automatically thresholded using a polynomial expansion technique to extrapolate the proper threshold [3]. The threshold exceedances are then screened further using a connectivity [3] process with several gate sizes. The connectivity algorithm has better performance than the LMS algorithm but is more difficult to implement for the entire image. Because of this, the LMS is used as a preliminary screening algorithm. The connectivity algorithm basically tests the detection to see if it is connected to other objects in the scene. The detection must be isolated in order to consider the object as a target. The detections that pass the connectivity screening are assembled into a track file utilizing frame-to-frame correlation of spatial location and intensity in the image. The track analysis module will analyze the track for merit and prioritization.

In some cases the sensor must look down to detect an aircraft. In this circumstance, there may be high clutter in the scene. The automatic thresholding module will sense the clutter content and inform the detection algorithm of this condition and a high clutter analysis will be done. In high clutter the target cannot be uniquely separated from the other target-like objects in the scene since they will have the same target-like signatures as the actual aircraft. When activated the high clutter module will lower the automatic threshold enabling many false targets with target-like signatures to pass the LMS and connectivity screening. The high clutter module will then analyze the detections for motion, local image context, and shape. If a potential

target has sufficient evidence to identify it as an aircraft then it will be reported as a detection. An example of aircraft detection in a high clutter scene is shown in figure 3.

A passive range estimate will also be performed using the angular motion of the potential target. For a moving platform this will be accomplished by having the platform execute a maneuver. The relative motion of the potential target before is compared with the relative motion after, and the target range is calculated using the change in velocity of the platform and assuming the target velocity did not change. For stationary platforms the rate of growth of the target coupled with the angular velocity of the target will determine the velocity direction. Knowing the direction, then a range assumption can be calculated for each class of aircraft by comparing the size of the target to the characteristic dimensions of the aircraft at that aspect.

AIRCRAFT IDENTIFICATION

Aircraft identification is accomplished by using imagery from the visual sensor during the day and from the IR sensor at night. When the size of the detected aircraft becomes large enough to be identified, the sensor will stare at the aircraft and an identification will be attempted. In order to identify an aircraft, the aspect and range are useful. This data enables a model based approach to be used which improves classification performance. The basic approach used for aircraft identification was motivated by the US Army "Visual Aircraft Recognition Manual" [4]. This manual serves as an Army training manual for aircraft recognition. The manual indicates that humans identify aircraft by looking for distinguishing characteristics of aircraft parts such as distinctive shapes or configurations of wings, fuselages, vertical stabilizers, or other aircraft parts. Based on this information, a knowledge based system was designed which utilized a parts analysis. A model generating module was developed for this system to provide references for the aircraft parts. This model generator can generate a crude silhouette for any of the aircraft classes for any aspect angle. The parts of the aircraft models can then be matched to the aircraft image using a parts matching technique. An aircraft class is declared based on the parts analysis. The knowledge based system consists of several modules. These modules are the model guided segmentation, aspect determination, model generator, and parts matching. The model guided segmentation module segments the aircraft based on a crude background model. The aspect determination module determines the aircraft pitch, yaw, and roll angles by using motion or a skeleton approach.

The parts matching module will match model parts to the aircraft parts. The model generator generates models of the aircrafts for the determined aspect. An outline of this system is shown in figure 4.

The segmentation procedure uses a region segmentation called the Histogram Optimization Segmentation (HOS) [5,6] (this is similar to edge preserving smoothing [7,8]) followed by a model guided merging [9]. The HOS algorithm applies an adaptive 3x3 filter which smooths areas of uniform grey level while enhancing edges. The filter is applied until it converges. At this stage a simple form of clustering which groups adjacent pixels that have an intensity difference below a segmentation threshold [5,6] is applied. This grouping creates the image segments. The segmentation threshold must first be determined to accomplish the segmentation. This is done automatically by resampling the image so that the anticipated target is only a few pixels in the image. Then the magnitude of the contrast of the adjacent points in the resampled image is determined and a contrast histogram is created. Based on the contrast histogram two trial segmentation thresholds are examined. Each one of these segmentations is identified with a hypothesis. One hypothesis assumes that the target has the largest contrast. Another assumes that the target contrast is essentially the contrast of the background objects. Each one of these trial segmentation thresholds is executed then a hypothesis verification is done. The verification is done by examining the size, spacial distribution, and number of segments in the trial segmentations. Characteristics such as these are used to confirm a successful trial segmentation. For example a high confidence is associated with the first trial segmentation if only one small segment is present after segmentation. Also a low confidence is associated with the first segmentation if many small segments are present in the trial segmentation. Similarly a high confidence is associated with the second trial segmentation if it has only one small segment.

After the HOS segmentation is accomplished, then a merging algorithm is used. The algorithm uses rules that are aimed at combining segments that are large and uniform throughout the image. This is done since the sky background is modeled as a large area of relatively uniform contrast changes. The merging algorithm merges the background pieces then labels the remaining segments as a target.

One advantage of designing a system to recognize aircraft is that once a target velocity vector is known then the yaw and pitch of the aircraft is determined. This is due to the observation that in normal flight, an

aircraft will fly with their major axis (the axis between the nose and tail) parallel to their velocity vector. Additionally, once their pitch and yaw are known then simple features (such as the width to length ratio and wing skeleton angles) can be used to identify the roll angle. The method used to determine the velocity vector is to measure the angular displacement of the aircraft in the image along with the change in length of the aircraft. These parameters are sufficient to determine the direction of the aircraft velocity vector for a stationary observer. For the moving observer at least the observer's velocity and the range to the aircraft need to be estimated or measured. To show that the velocity direction can be found for the aircraft using image motion, consider that the aircraft velocity is given by the following equation:

$$\underline{V} = \frac{dr}{dt} \hat{r} + R \frac{d\theta}{dt} \hat{\theta} + R \frac{d\phi}{dt} \hat{\phi} \quad (1)$$

where r , θ , and ϕ are the sensor pointing angles to the aircraft and seen in figure 5. Assuming that the velocity is uniform over a time Δt then the change in the aircraft length in the image (ΔL) is given by:

$$\Delta L = L \frac{\Delta R}{R} \quad (2)$$

so

$$\frac{\Delta R}{\Delta t} = \frac{R}{L} \frac{\Delta L}{\Delta t} \quad (3)$$

where L is the original length and ΔR is the change in range. Using the above in equation (1) yields:

$$\underline{V} = \frac{R}{\Delta t} \underline{q} \quad (4)$$

$$\underline{q} = \frac{\Delta L}{L} \hat{r} + \Delta \theta \hat{\theta} + \Delta \phi \hat{\phi} \quad (5)$$

Since the vector \underline{q} is parallel to \underline{V} the direction of \underline{q} is the direction of the velocity vector. The angles θ and ϕ can be computed by finding the change in location of the aircraft through images separated by time. The change in length of the aircraft can also be computed through time by computing the change in the major axis length of the target segment. Figure 6 shows a sequence of images where the direction of the velocity vector is computed in this manner.

In addition to using motion to determine aspect, the use of a type of skeleton [10,11] matching has been investigated for this problem. This technique uses a pseudo-skeletal analysis which differs from the conventional medial axis transform.

The aircraft is decomposed into a skeleton then the skeleton is matched to skeletons of the models at different aspects. The aspect of the best matching model skeleton is used as the aircraft aspect. This aspect determination is done for each class of objects. The method used is to limit the angle search to two angles. The approach used to generate the pseudo-skeleton is to initially find the longest axis that can fit in the segmented aircraft. Then appendages that extend from this axis are found (see figure 7). The major axis is derotated to horizontal (this derotation angle is either the pitch or roll of the aircraft depending on the matching skeleton). The derotated skeleton is then matched to a set of stored skeletons for a particular class to find the yaw and roll or the yaw and pitch. The matching is accomplished using cost functions [12].

A model based (top-down) approach [13] is used to match the model aircraft parts to the segmented aircraft. The classifier carries the capability to generate a reference silhouette for a model for any aspect. The models are stored as a set of 3-dimensional points. Each point is the corner of a planar surface. A complex representation was not adopted since simple models seem to be effective and require fewer computations to generate [9]. The surfaces were determined by examining aircraft outlines contained in Reference 4. The vertex points were transferred according to the pitch, yaw, and roll angles which are determined from the aspect determination module. An example of the model generation for various aspect is shown in figure 8. Models for the A-7, F/A-18, F-5, F-111, MiG-25, B-52, Tu-26, F-15, and MiG-21 aircraft are stored. Additionally each surface is labeled according to its aircraft part.

A parts matching is used to classify the aircraft. This parts matching is done in a top down manner based on the identified parts of the class models. A match is done on the nose, body, wings, vertical stabilizer(s), and horizontal stabilizers. A present time a simple area match confidence is used. The technique is to size the generated model so that its area is equal to the segmented aircraft, then align their centroids. A confidence of match is determined for each part of the model by computing the ratio of the intersection of the model part and the segmented aircraft to the area of the model part. In this manner a confidence measure is determined for each of the different model's parts. A simple rule system then makes a classification decision based on the confidence measures and the size of the model parts. The system declares the aircraft to be a particular class if that class has the best match and all

its visible parts match with good confidence. Allowances are made for part matches where the part is small and therefore can be easily lost during segmentation or imaging.

IDENTIFICATION COMPARISON RESULTS

Two statistical classifiers were compared to the knowledge based classifier in order to determine if it was more accurate than conventional classifiers. These statistical classifiers were near neighbor classifiers that used the same segmentation, aspect determination, and model generation as the knowledge based classifier. However the statistical classifier would classify based on features such as Hu's moments or discrete fourier transforms of the boundary. These features were stored as a function of aspect. The features were chosen based on maximizing the F-statistic through the training data. In this analysis two groups of features were found that had essentially equivalent F-statistics so two classifiers were used with each one using one of the feature groups. The classifiers were compared using real imagery and using a controlled test. When real imagery was used the knowledge based classifier performed much better than the statistical classifier, however the performance difference was a strong function of the test data so a more controlled test was needed. A controlled test was accomplished where noise of varying amplitude was added to aircraft silhouettes then the degraded silhouettes were segmented and run through the classifiers. An example of adding the noise is shown in figure 9. The results of this comparison are shown in figure 10. It can be seen that, for the four aspects, the knowledge based classifier performed significantly better than the two statistical classifiers.

Additional tests also performed to analyze the performance of the knowledge based classifier to the statistical classifiers for different range targets. In this test targets were simulated for different ranges by undersampling the resolution (see figure 11). Other effects such as atmospheric effects were not into the model so as to isolate the effect of spacial sampling on the different classifiers. The comparative results for different ranges are shown in figure 12. It can be seen that the knowledge based classifier performed significantly better than the statistical classifiers.

CONCLUSION

The synergistic approach to using the information from both the detection and identification systems will increase the performance of both systems. The identification is greatly assisted by being provided a passive

range and velocity vector of the aircraft. This allows a model based approach to be used which as this study shows has a much better performance over the conventional classifiers. The classifier can also assist detection by providing target and clutter information which can help the detection process determine the optimal detection threshold.

REFERENCES

1. "Least-mean-square Spatial Filter for IR Sensor," E. Takken, D. Friedman, A. Milton, and R. Nitzberg, *Applied Optics*, Vol. 18, No. 24 (Dec 1979) pp. 4210.
2. "Simulation of Mid-infrared Clutter Rejection," M. Logmire, A. Milton, and E. Takken, *Applied Optics*, Vol 21, No. 21 (Nov.1982)
3. "A Comparison ofIRST Algorithms For Imaging Systems", G. Roberts, L. Voelz, and D. Risch, *proceedings National IRIS*, 1987.
4. "Visual Aircraft Recognition", US Army field manual FM 40-30, October 1983 pp. 3819.
5. G. Roberts, "Histogram Optimization and Compression Segmentations", *SPIE Applications of Imaging Tech*, Vol.397, April 1983, pp. 487-493.
6. G. Roberts, "Expert System for Labeling Segments in FLIR Imagery", *SPIE Application of Artificial Intelligence IV*, Vol 635, April 1986.
7. M. Nagao and T. Matsuyama, "Edge Preserving Smoothing," *Proc. 4th International Joint Conf. on Pattern Recognition*, Kyoto, Japan, Nov. 1978.
8. F. Tomiti and S. Tsuji, "Extraction of Multiple Regions in Selected Neighborhoods," *IEEE Trans. Systems, Man, Cybern.*, Vol SMC-5 (1977), pp. 107-109.
9. G. Roberts, "Model Guided Segmentation of Ground Targets", *SPIE Application of Artificial Intelligence V*, Vol. 786, May 1987, pp. 198-205.
10. H. Blum, "Biological Shape and Visual Science," *J. Theoretical Biology* 38, 1973, pp. 205-387.
11. D. Marr, "Representing Visual Information," *AI Memo 415*, AI Lab MIT, May 1977.
12. M. Fisher and R. Elschlager, "The Representation and Matching of Pictorial Structures," *IEEE Trans. Computers* 22, 1, Jan 1973, pp. 67-92.
13. J. Tenenbaum and H. Barrow, "Experiments in Interpretation Guided Segmentation," *Artificial Intelligence*, 8, (1977), pp. 241-274.
14. M. Hu, "Visual Pattern Recognition by Moment Invariant," *IRE Trans. Info. Theory*, Vol IT-8, pp. 179-187.

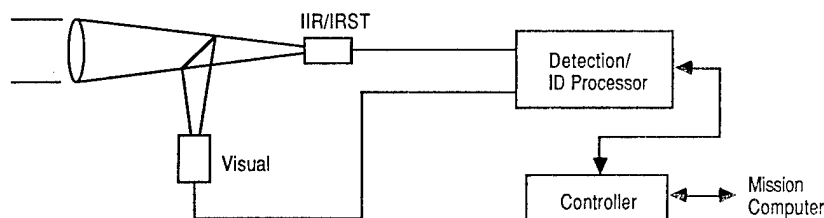


Figure 1; The detection and identification system using a IIR/IRST sensor and a visual sensor.

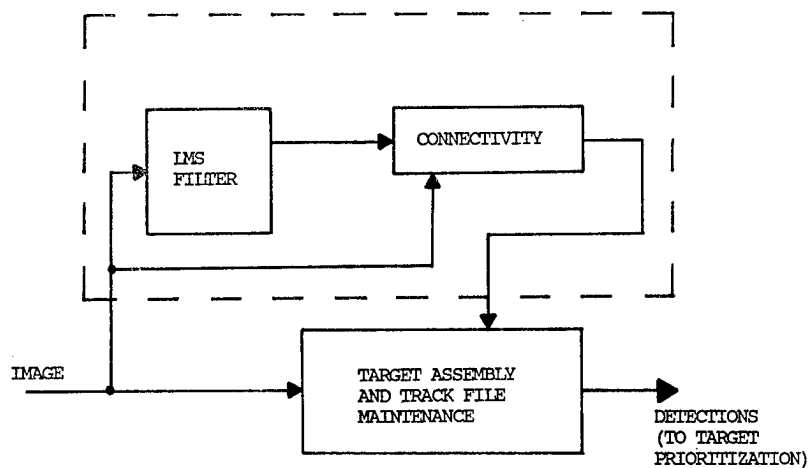


Figure 2; The detection algorithm structure.

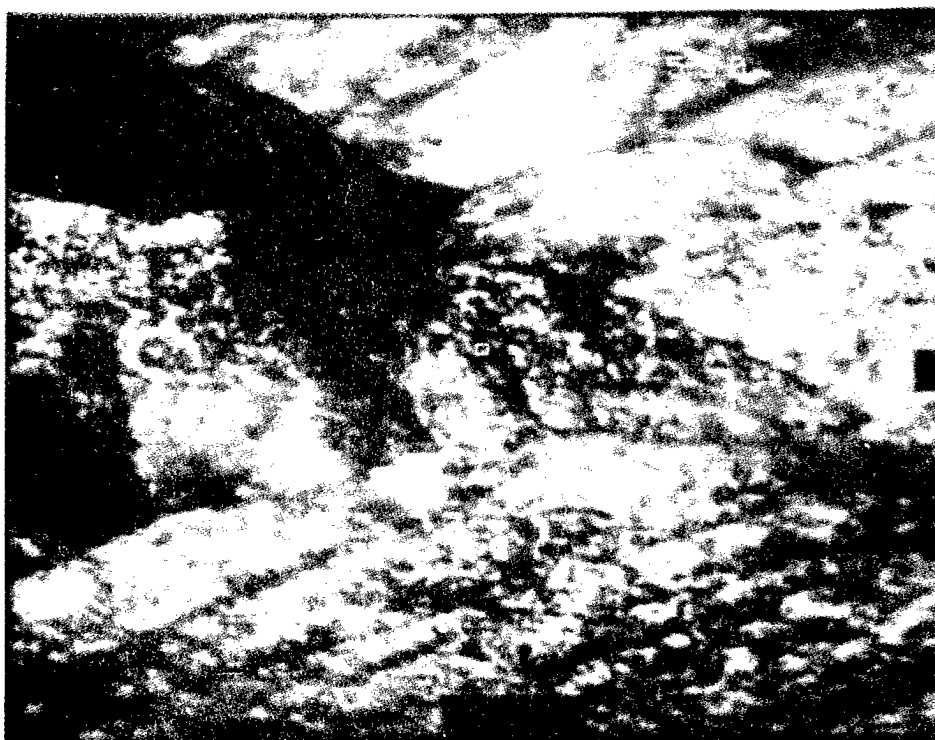


Figure 3; The detection without false alarms of the moving aircraft (boxed target in center of image) in high clutter.

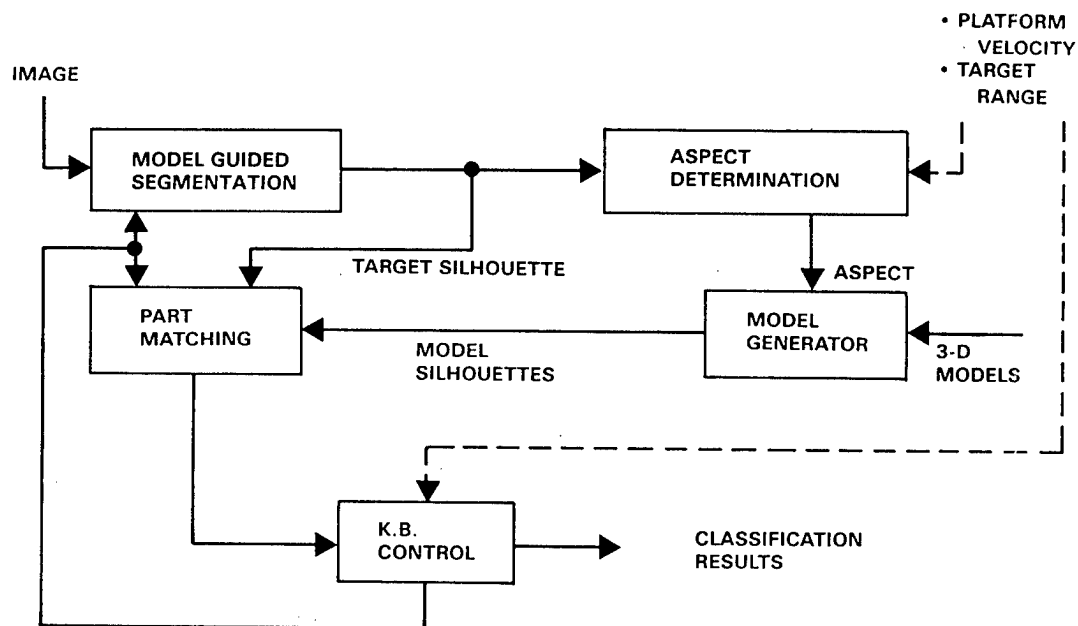


Figure 4; Outline of the knowledge based system for aircraft recognition.

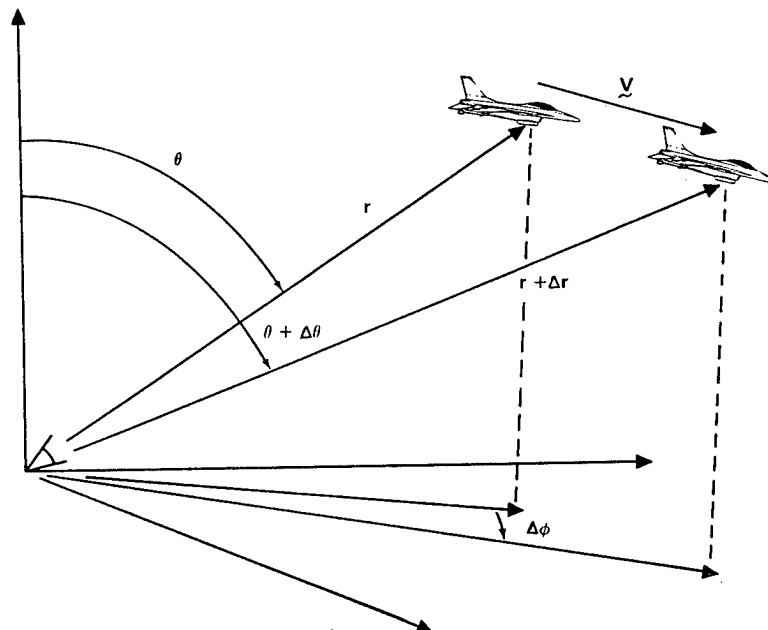


Figure 5; The pointing angles to an aircraft (θ and ϕ) and the range (R).

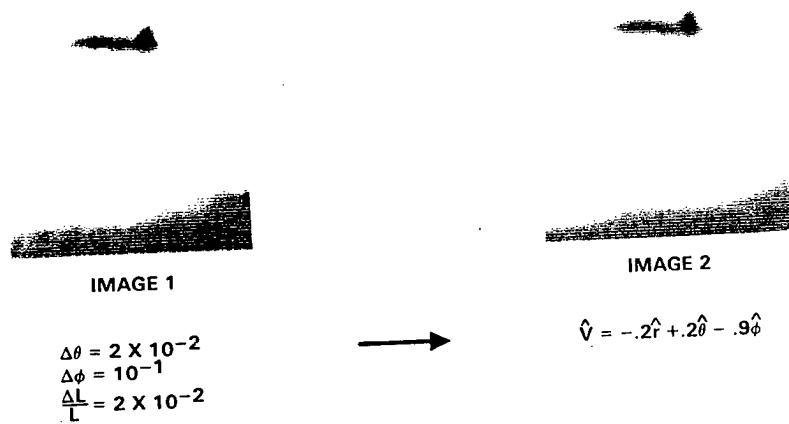


Figure 6; The velocity vector computed from interframe motion. The image displacement and change in length of the aircraft is used to determine the direction of the velocity vector.

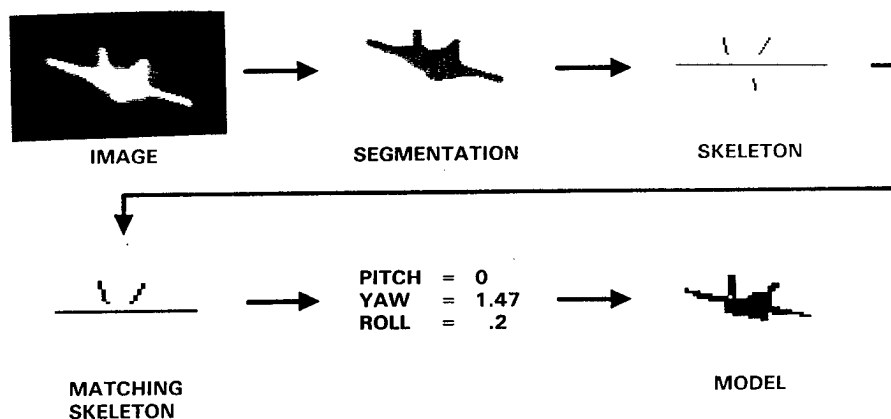


Figure 7; The skeletal matching procedure. An image is segmented, then the major axis and the skeletons is derotated and matched to models thus finding the roll angle.



Figure 8; Various aspects of a Tu-26 (Backfire) bomber generated with the model generator.

UNCLASSIFIED



Figure 9; An example of degrading a target silhouette with noise.

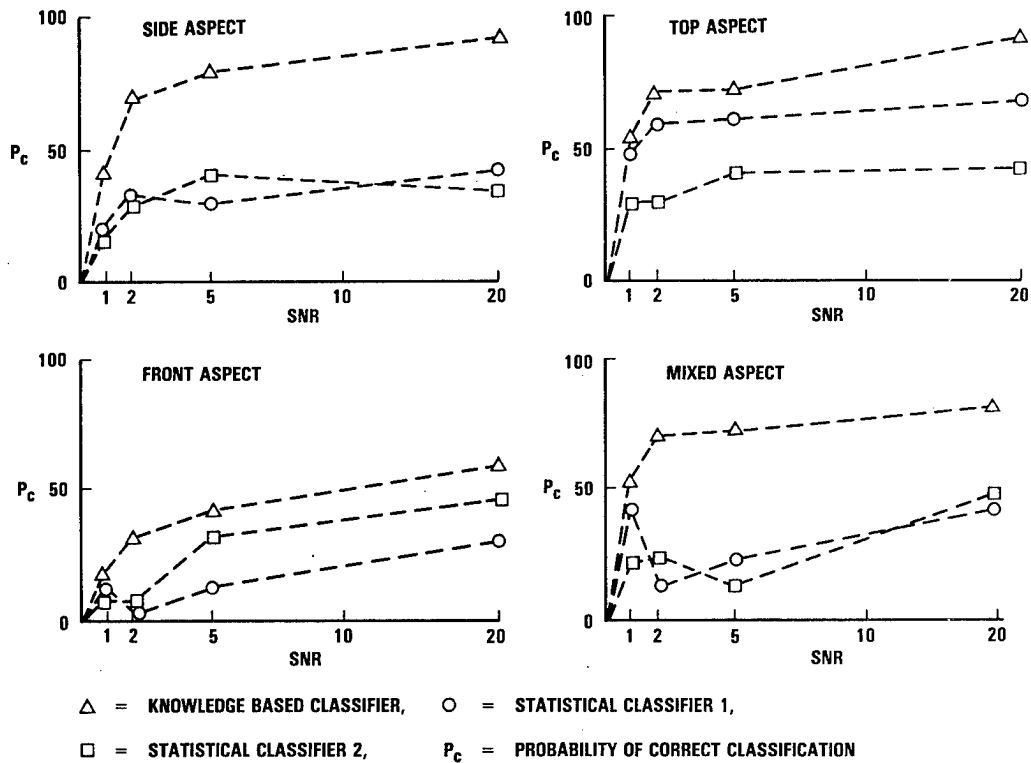


Figure 10;The comparative probability of correct classification for the knowledge based classifier and the statistical classifier as a function of signal-to-noise ratio (SNR).

UNCLASSIFIED

UNCLASSIFIED

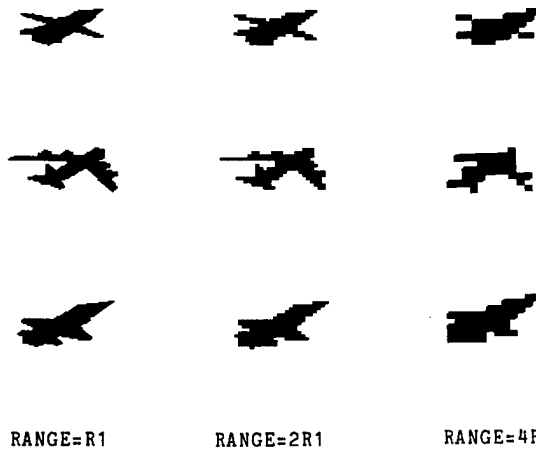


Figure 11; Examples of the spacial sampling degrading of a aircraft at various ranges.

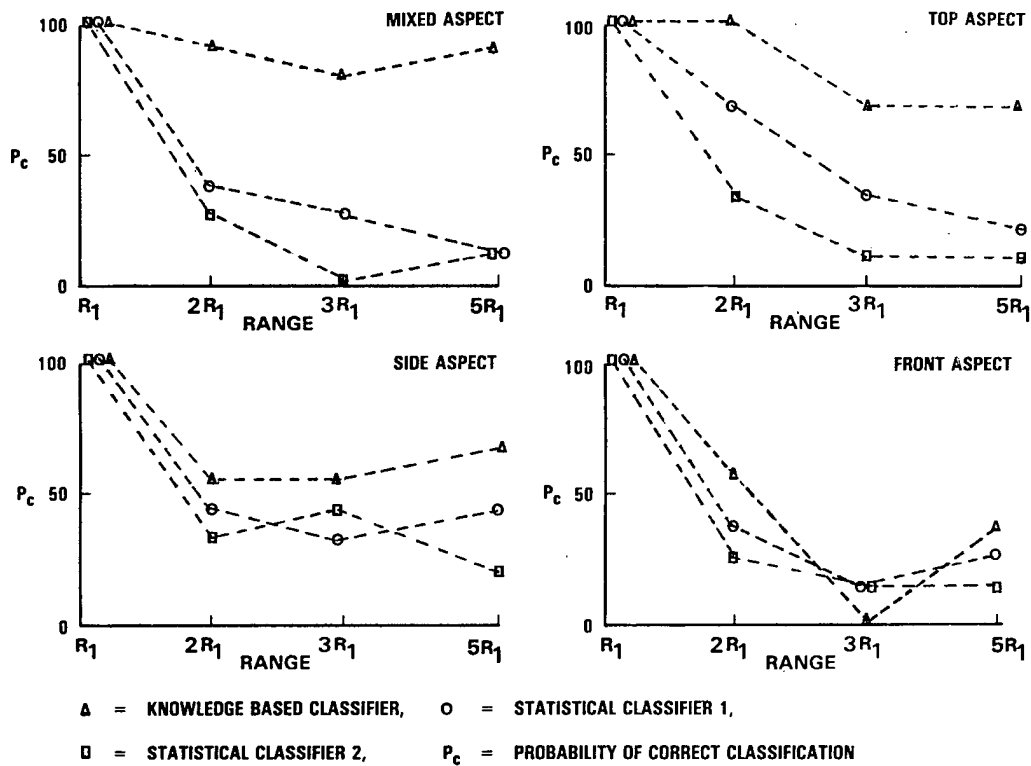


Figure 12;The comparative probability of correct classification for the knowledge based classifier and the statistical classifier as a function of spacial degradation range.

UNCLASSIFIED

THIS PAGE LEFT BLANK INTENTIONALLY

USE OF A TRACK-TO-TRACK ASSOCIATION ALGORITHM
IN THE FUSION OF LONG-RANGE SENSOR DATA

David H. Kaskowitz
Robert G. Bryson
William H. Barker

TIBURON SYSTEMS, INC
2085 Hamilton Avenue
San Jose, CA 95125

ABSTRACT

A computer-based algorithm is described that performs the fusion of ship tracks from different sensors or data sources but which pertain to the same platform. It is applicable to long-range sensor data for the Over-the-Horizon Targeting (OTH-T) function. As a component of a Tactical Data Processor (TDP), the algorithm conducts pairwise comparison of tracks on the basis of positional statistical distance. Initial screening, based on some attributes, age of data, and position feasibility, filters out candidate tracks that are clearly unrelated or of no interest. Evaluation of the algorithm to date shows an advantage over manual fusion both in process time and capability to find track merge candidates.

INTRODUCTION

Over-the-Horizon Targeting (OTH-T) requires the correlation and fusion of long-range sensor data from a number of different sources. Because of the characteristics of these sensors and sources, it frequently happens that a given ship platform is represented by two or more constructed tracks corresponding to different sensors or sources.

In order to produce an accurate OTH-T picture, it is necessary to identify and merge these tracks together. This function is usually referred to as track-to-track association. Currently, there is no automatic track-to-track association function in any of the existing OTH systems. With the development of new sensors and sources of OTH-T data, such a function is increasingly important.

Tiburon Systems has developed an experimental track-to-track association algorithm, based on a Kalman-filter tracking algorithm called the Maneuvering Target Statistical Tracker (MTST). The initial development work was performed as part of an independent research and development project within Tiburon (Ref. 1). Additional work in this area has been funded by the Naval Underwater Systems Center (NUSC), Cruise Missiles Project Office (CMPO), and the Space and Naval Warfare Systems Command (SPAWAR).

The algorithm performs pairwise comparisons among tracks in a data base, and identifies tracks that are recommended to be merged. The algorithm includes initial screens to detect tracks that are incompatible on the basis of their attributes, age difference, and positions. A track pair that passes the initial screens is subjected to a detailed positional analysis which results in a decision whether to reject the track pair as merge candidates, to include the track pair as merge candidates, or to defer the decision until more data are available.

BASIS OF THE MEASURE OF ASSOCIATION

The measure of association is based on the average of the statistical distances between contacts of a comparison track to the filtered, smoothed solutions of the reference track at the times-of-event (TOEs) of the contacts. This approach is an extension of the concept of testing for an outlier contact in a single track, as described in Reference 2 regarding contact reliability. In this approach, the contacts (excluding the contact being tested) are used to forward

filter and backward smooth the track. The statistical distance is computed between the contact and the smoothed solution at the contact time-of-event. This is the "jackknifed" solution. The measure has an M degrees-of-freedom Chi-square distribution, where M is the dimension of the position vector (M = 2 for position only, and 4 for a full state report). If the test statistic has a small probability of occurrence, the contact is deemed an outlier. The test has been shown in Reference 3 to be equivalent to a likelihood ratio test of the null hypothesis, that the contact is a member of the track, against the alternative hypothesis, that the contact is displaced from the smoothed track. This is the "slippage" model.

In the track-to-track association algorithm, the contacts of one track are tested individually against the smoothed "reference" track. This may be viewed as a multi-dimensional "jackknifed" or a "slippage" model. The quadratic statistical distance is defined as follows:

$$\tau(k) = [Y(k) - \mu(k|n)]' * [\Sigma(k, k|n) + R(k)]^{-1} * [Y(k) - \mu(k|n)]$$

where

Y(k) is the position of the kth candidate contact,
 $\mu(k|n)$ is the smoothed solution of the reference track (having n contacts) at the TOE of Y(k),
R(k) is the position covariance of the kth candidate contact, and
 $\Sigma(k, k|n)$ is the position covariance of the smoothed solution of the reference track at k.

In the track-to-track measure, N such statistical distances are computed, one for each contact of the candidate track over a sequence of contacts {JK} that lie within the time span of the reference track when time overlap is present, or the end contacts when the tracks are non-overlapping in time. The measure, TAU, is the distance, $\tau(k)$, averaged over $k = JK(1), \dots, JK(N)$.

If the contact is a member of the track, which is the null hypothesis, the distribution of TAU will be Chi-square with M degrees of freedom for N = 1; for N > 1, the distribution must be found empirically because the N measures are not independent.

Based on the statistical distribution of TAU, a threshold may be established to reject a track pair as

pertaining to the same platform. However, the test cannot effectively discriminate two tracks of platforms traveling in "parallel" where the separation distance is in the same order of magnitude as the areas of uncertainty (AOU) of the sensor measurements. A conservative approach to thresholding for accepting the track pair for merging, postulates an alternate hypothesis that assumes the contacts to be from a track that is "parallel" to the reference track at a distance d. This simulates a "worst case" alternative hypothesis. In this case, for N = 1, the distribution will be a non-central chi-square with a non-centrality parameter based on the distance d. For N > 1, the distribution will be empirical based on a non-centrality parameter defined as follows:

$$\lambda = (d^2 / \sigma_k^2)$$

averaged over $k = JK(1), \dots, JK(N)$

where

d is the minimum separation distance between parallel paths that is to be discriminated, and

σ_k is the standard deviation of the joint distribution of position uncertainty of the two tracks at the TOE of the kth contact, i. e.,

$$\sigma_k^2 = (A + \sqrt{A^2 - 4B}) / 2$$

where

A = trace of (R(k) + $\Sigma(k, k|n)$)

B = determinant of (R(k) + $\Sigma(k, k|n)$)

The probability density function of TAU is approximated by the gamma distribution

$$g_i(x) = \frac{\alpha(i)}{\Gamma(r(i))} [\alpha(i)x]^{r(i)-1} e^{-\alpha(i)x}$$

where i = 0, 1 pertains to the null and alternate hypotheses, respectively

$\Gamma(r(i))$ is the gamma function of r(i)

$$\alpha(i) = \mu(i) / \sigma(i)^2$$

$$r(i) = \mu(i)^2 / \sigma(i)^2$$

and the parameters $\mu(i)$ and $\sigma(i)$ are as follows:

Parameter	(i=0) NULL HYPOTHESIS	(i=1) ALTERNATE HYPOTHESIS	
		N > 1	N = 1
$\mu(i)$	=	$\beta_{0m} + \beta_{1m} \lambda$	$1 + \lambda/2$
$\sigma(i)$	=	$(\beta_{0s} + \beta_{1s} \lambda) / \sqrt{N}$	$\sqrt{1 + \lambda}$

where C_1 is the complexity of the reference track. Complexity is defined as the ratio of the distance from the first to the last contact on the track to the sum of the distances between adjacent contacts. If the ratio is less than 0.7, the track is complex and $C_1 = 1$; otherwise, the track is simple and $C_1 = 0$. The coefficients, β , are derived experimentally and are a function of the distance parameter, d , (in nm) and the complexity C_1 , as is shown in Table 1.

that have new contacts added or deleted, or that have been merged with another track. The Correlator Tracker processes all constructed tracks using the Maneuvering Target Statistical Tracker (MTST) of Reference 2. The purpose of this filtering is to remove outlier (aberrant) contacts from the track, and to provide a smoothed track based on the "active" contacts that passed the outlier testing. The original contact data, the smoothed track data, and the variables that

TABLE 1
VALUES OF EMPIRICAL COEFFICIENTS

COMPLEXITY	SYMBOL	$d < 1$	$1 \leq d \leq 3$	$3 < d \leq 5$	$5 < d \leq 10$	$10 < d$
$C_1 = 0$ (SIMPLE)	om	1.82	1.81	1.80	1.86	2.31
	lm	0.00	1.75	1.70	1.65	1.61
	os	2.30	2.44	2.76	3.79	5.48
	ls	0.00	1.73	1.48	1.36	1.92
$C_1 = 1$ (COMPLEX)	om	2.95	2.71	2.36	1.08	-1.62
	lm	0.00	2.05	2.00	1.95	1.94
	os	4.36	4.52	4.07	2.88	0.46
	ls	0.00	1.03	1.42	1.61	1.65

The M-score thresholds are based on the type I and II errors of hypothesis testing: type I is the probability of rejecting the null hypothesis when it is true, and type II is the probability of accepting the alternate hypothesis when it is false. The thresholds are based on the incomplete gamma functions $G(x; r, \alpha)$:

The R-score is the probability of making a type I error:

$$R\text{-SCORE} = 1 - G_0(\text{TAU}; r(0), \alpha(0))$$

The M-score is the probability of making a type II error:

$$M\text{-SCORE} = 1 - G_1(\text{TAU}; r(1), \alpha(1))$$

The thresholds of the R-score and M-score are R_0 and M_0 , respectively. They are input control parameters. The thresholding is as follows: If $R\text{-score} < R_0$, the candidate track is rejected as belonging to the reference track. If $M\text{-score} = M_0$, the candidate is recommended for merging with the reference track. Otherwise, the candidate is retained for further testing.

DESCRIPTION OF ALGORITHM

The Track Association Algorithm (TAA) is configured to be a component of a Tactical Data Processor (TDP). Figure 1 shows a system in which the identified blocks are functions performed by the TDP.

On the input side, the TAA Input Control initiates a cold start/warm start, and manages the Track Update Queue to the TAA. Track updates may consist of tracks that are deleted,

enable the smoothed track to be interpolated and extrapolated, are stored in the Tracker Support File. In conjunction with the TDP Data Base Management System (DBMS), the track attributes are stored in the Attribute Data File. The TAA Control Parameters Man Machine Interface (MMI) permits the adjustment of a set of control parameters.

On the output side, the TAA produces Merge Recommendation Data, a Candidate List, and an Operator Reject List. The TDP utilizes these data to make track merge decisions and resolve conflicting merge recommendations. A typical conflict exists when track A is recommended for merging with tracks B and C, but where tracks B and C have been rejected as the same platform. The TAA Operator Query may request that a Console Selected track (CST) be evaluated versus all candidates, and the candidates ranked in order of their M-scores. An ELINT Support Data Base may be used to help resolve merge recommendation conflicts, or evaluate choices among multiple merges, by correlating ELINT parameters with non-ELINT attributes at a level higher than that performed in the attribute screening. The Merge Processing Man Machine Interface (MMI) is the function used by the operator to decide which tracks to merge; a decision to merge two tracks is conveyed to the Data Base Management System (DBMS) which will perform the track merging and update the Track Update Queue. The Merge Processing MMI may also update the Operator Reject List (as a result of conflict resolution and merge decisions) which causes the TAA to exclude the track in question from the candidate list and future merging. A Status Query option is available so

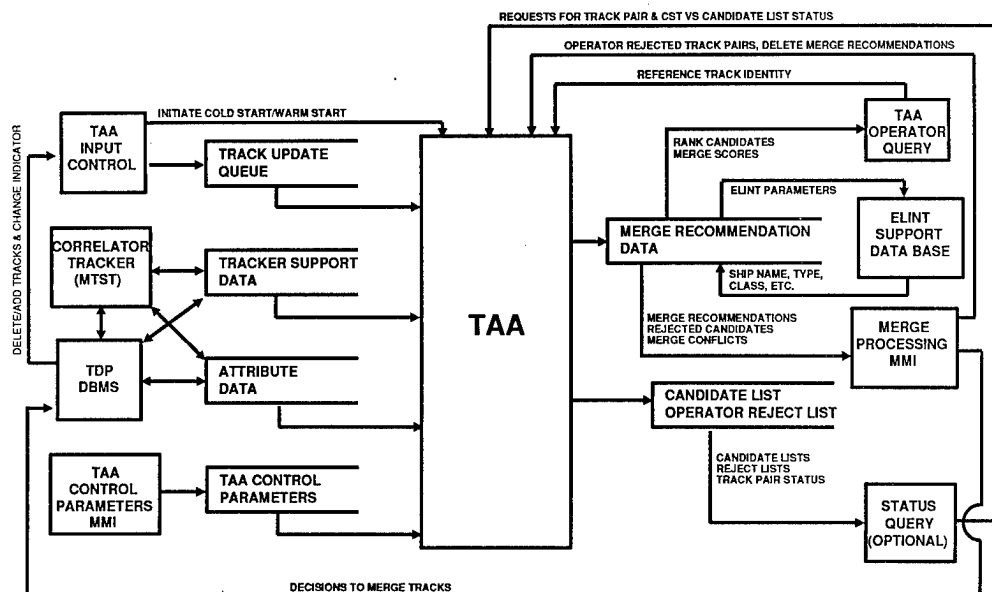


Figure 1

that the TDP may examine the status of the Candidate List and Operator Reject List.

Figure 2 shows the block diagram of the TAA. The four major functional modes are shown. The Compare mode is the principal mode. The algorithm automatically compares each track in a source list to all tracks in a candidate list. Initially, the candidate list is determined by the source(s) of the contact data in the track. After the first pass, the candidate lists contain only candidates that have not been rejected. Subsequently, however, a rejected track may be reinstated in the candidate list if the TDP has determined that the track has changed. A change occurs if any of the specified MTST parameters has changed, if any active contact from the earlier track is deleted or becomes inactive (except the eldest contact), or if there is a position change in one of the previously active contacts.

For an initial pass of any track, the Screen Candidate function is performed. A candidate may be rejected at this level on the basis of attribute inconsistencies and position infeasibility. Candidates, whose age difference with respect to the reference track is outside a region of interest, are retained in the candidate list but are not further analyzed.

The measure of association is evaluated for candidates which pass the screening tests, and provided that at least one of the tracks in the pair have multiple contacts. The measure is

computed for one track as the reference track, and the candidate as the comparison track, and then with the two tracks in reverse roles. An exception occurs when either track contains only

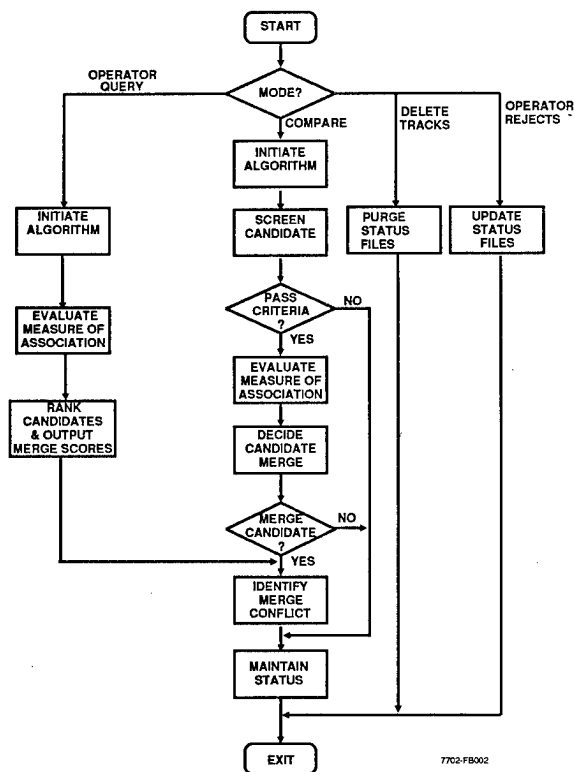


Figure 2

a single contact. In this case, only the multi-contact track may serve as a reference track because the expansion of a single contact is judged to be unreliable. If two R-scores and M-scores are obtained, they are not necessarily symmetric.

The logic for making the decision to recommend a merge is as follows: The candidate is rejected if one of the R-scores has failed to pass the reject threshold. If otherwise, the candidate is recommended for merge if one of the M-scores passes the merge threshold and provided that both tracks are multi-contact. Otherwise, the candidate is placed on hold.

Merge recommendations are recorded in the Merge Recommendation Data file. Cross checking the merge data is performed to identify and record merge recommendation conflicts.

The Operator Query mode evaluates the M-scores for a given CST versus its candidate list which may or may not exclude tracks derived from specified sources. The candidates are ranked in order of their merge scores even if the merge scores do not pass the merge threshold. The maximum M-score is used in the ranking. For candidates that are recommended for merge, the merge conflicts are identified if existent.

The Delete Tracks mode purges the Candidate Lists and Merge Recommendation Data of tracks that have been deleted from inventory by the TDP. The Operator Reject mode updates the above data files to enforce merge decisions made by the TDP operator.

DISCUSSION

The performance capability of TAA has been tested against ground truth based on simulated track data. The test compared the algorithm's performance to that of several operators. A series of tests applied to archived operational data is presently under way.

An ensemble of 31 tracks was installed on the Shore Targeting Terminal System, a derivative of the AN/USQ-81(V). The smoothed tracks were plotted on the geo display, and a series of view foils were made of the smoothed tracks and their covariance ellipses ("slinkies"). Figure 3

illustrates a sequence of position contacts and the contact areas of uncertainty (AOUs). In Figure 4, the contacts of Figure 3 have been smoothed by the MTST filter. All the tracks in the ensemble were in the same time period so that time overlap was always present. Attributes were excluded in the test. These data and the contact reports were given to operators. They were asked to pick out tracks that represented the same ground truth. The results of their performance are given in Table 2.

The same ensemble of tracks was evaluated by the TAA algorithm. The separation distance criterion d , was varied over the set (5,10,15,20) NM, and the MERGE THRESHOLD, Mo, was given two values, viz., 99 and 98 per cent corresponding to Type II errors of 1 and 2 per cent, respectively. The performance results are shown in Table 3.

As is shown in Table 3, an increase in the distance parameter, d , improves the detection of track pairs that should be merged, but at the expense of increasing the number of incorrect merges. The latter represents the Type II error. A conservative approach especially applicable to automatic processing, where manual evaluation is to be maintained at a minimum, mandates that the rate of incorrect merges be small, while depending upon future evaluations (as the tracks are updated) to correct the missed merges. A low rate of incorrect rejections, the Type I error, is required so that the missed merge tracks will automatically be resubmitted for evaluation upon their updating. For the test ensemble above, the incorrect rejection rate was nil.

TABLE 2
PERFORMANCE OF OPERATOR IN CORRELATING 31 TRACKS

OPERATOR NUMBER	EXECUTION TIME (HRS)	CORRECT MERGES	MISSED MERGES	INCORRECT MERGES
1	1/4	7	17	6
2	2/3	15	9	4
3	1-1/3	4	20	17
4	4	18	6	11

TABLE 3
PERFORMANCE OF ALGORITHM IN CORRELATING 31 TRACKS

SEPARATION CRITERIA	MERGE THRESHOLD	CORRECT MERGES	MISSED * MERGES	INCORRECT MERGES
5 NM	99 %	10	14	0
10 NM	99 %	16	8	0
15 NM	99 %	19	5	3
20 NM	99 %	24	0	8
5 NM	98 %	13	11	0
10 NM	98 %	16	8	1
15 NM	98 %	21	3	3
20 NM	98 %	24	0	8

* HOLDS

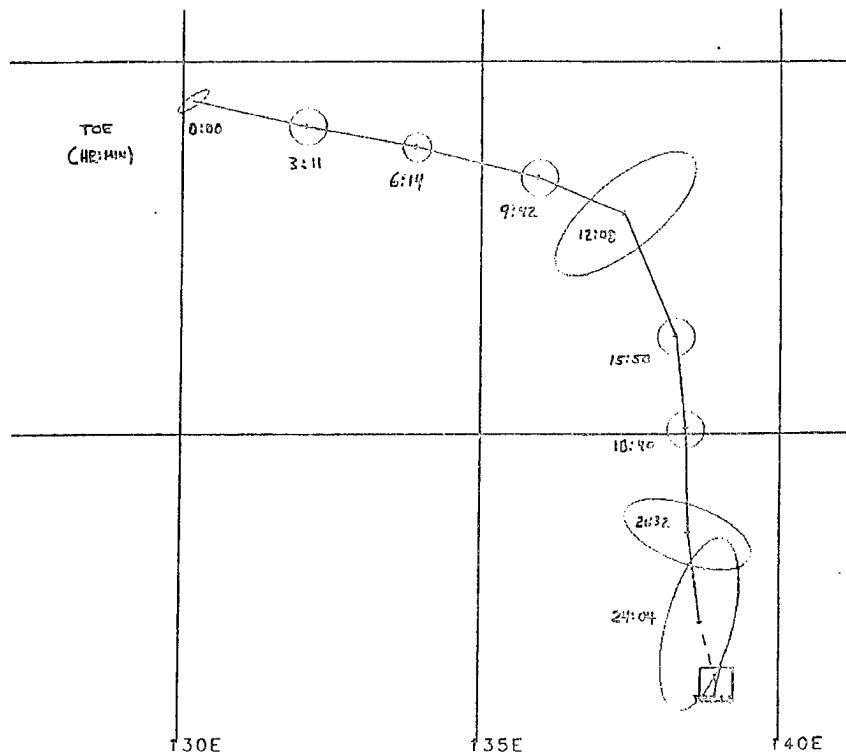


Figure 3

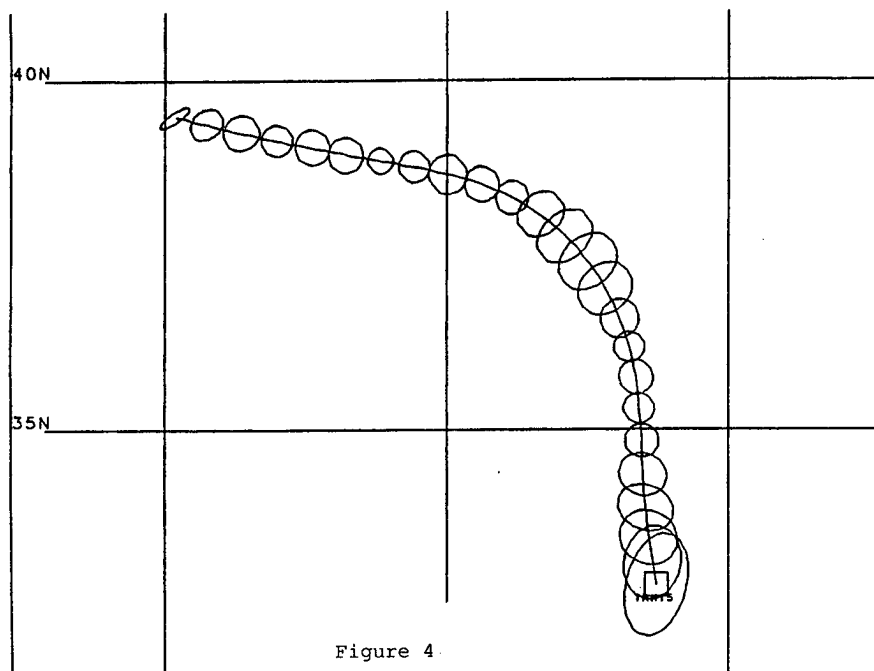


Figure 4

CONCLUSIONS

The need for an automatic track association algorithm in current TDPs has been demonstrated. With as few as 30 tracks in a data base, an operator will be faced with over 400 possible pairwise comparisons. With a larger data base, the task is overwhelming for manual solution.

The algorithm that was developed is based on a statistical comparison between two tracks. In contrast with other approaches, it uses multiple contact data of a track.

The results concerning the performance of the algorithm indicate that it is effective in filtering out pairs of tracks that come from different platforms. However, there still appears to be a need for an operator to make the final decision whether two tracks should be merged.

The TAA approach to implementation of track association can be seen as an alternative to the Bayesian multiple hypothesis systems that have been proposed. The TAA approach assumes that a series of correlator-trackers have operated on each of several input data streams to form tracks. The function of TAA is to sort through the sets of tracks and identify those that are merge candidates. The advantage of the TAA approach includes its implementation in a modular form in a TDP, exploiting the correlation algorithms that are in place.

ACKNOWLEDGEMENT

The authors extend thanks to Trang Nguyen for generating the statistical data cited in the paper.

REFERENCES

1. "Track to Track Association Final Report", Tiburon No. 02329, September 29, 1985, Tiburon Systems, Inc., San Jose, CA
2. "Over the Horizon/Detection Classification and Targeting (OTH/DS&T), System Level Specification, Ship Tracking Algorithm", PDE-120-S-00454, REV A, MAY 1985, Naval Electronic Systems Command, Command and Control Systems, Directorate PDE-120, Washington, D. C. 20363
3. W. H. Barker, "The Relationship Between Contact Reliability and Other Statistical Tests for Outliers in Tracking/Filtering Problems", Daniel H. Wagner Associates Memorandum, Final Revision, February (1983)

THIS PAGE LEFT BLANK INTENTIONALLY

A USER-SYSTEM INTERFACE DESIGN TOOL

Stephen L. Morgan and Alexander Nauda

HRB-Singer, Inc.
Box 60, Science Park
State College, PA 16804
(814) 238-4311

Described is a sophisticated system design tool for rapid prototyping and development of man/machine interfaces tailored to the needs of C³I, I&W, situation analysis, and other data fusion systems. The User System Interface Design Tool (USIDT) is a versatile, interactive software system which resides on a Silicon Graphics workstation with an external map database and scenario data files. With this advanced prototyping capability, designers can develop and demonstrate detailed interactive displays, menus, maps, and processing sequences early in the software design phase. USIDT capabilities include a robust display/menu construction tool set, a map database and scenario file retrieval system, a friendly interface for inexperienced users, and a strong simulation capability to allow end users to examine and critique the menu interaction, interactive map and data overlays, and zooming on the map displays (before any production software is generated). Further, USIDT provides the capability to collect, analyze, and report human factors data not available elsewhere in such a framework.

Prototyping the user system interface allows government reviewers to examine proposed system designs, to suggest changes to the structure or flow of the interaction, and to observe the results of the modifications. The more precisely a design can be shown to reviewers, the more likely that problems will be identified and corrected early in the software development life cycle, thereby reducing total system cost and producing a usable system, adhering to military standards. The use of this design tool will improve the user system interfaces of data fusion systems and will reduce development costs by minimizing late software life cycle rework as a result of the improved communication between system designers and government reviewers early in the design phase, when modification of the design is easier to accomplish and less expensive.

The use of USIDT is demonstrated through sample menus and graphic output. Simulation and human factors data collection capabilities are illustrated with examples of selected display sequence simulations.

A USER-SYSTEM INTERFACE DESIGN TOOL

Stephen L. Morgan and Alexander Nauda

HRB-Singer
Box 60, Science Park
State College, PA 16804

INTRODUCTION

With the availability of high speed data links, tracking sensors, airborne reconnaissance devices, satellite data, and recent advances in EW, there is a staggering amount of information available to the C³I, I&W, and situation analysis community. Although high speed computers can rapidly process, catalog, file, and format this data, it is the user interface that is responsible for presenting this information clearly and concisely to tactical and strategic decision makers (Ref. 1). While the availability of color CRT or raster scan hardware has created new options for the display of complex data, the flexibility offered by software systems makes possible the optimization of these displays to meet user requirements. However, the needs of the user frequently have been overlooked until the overall system has been developed, leading to increased costs due to rework late in the software life cycle. This, and the increased size and complexity of data fusion systems, has resulted in software costs exceeding hardware costs by an increasing ratio (Ref. 2,3). Industry experience indicates that the use of software development tools can improve productivity by identifying unsatisfactory elements (for example, in the man/machine interface design) early in the design process before any production code is generated, rather than later in the process when rework and reimplementation forces costs to be higher and results in delayed development schedules (Ref. 4,5). Without thorough consideration of the way data is displayed to decision makers, a data fusion system may be impotent. Rapid prototyping and simulation of user-system interfaces makes possible the evaluation of design trade-offs and engineering of human factors into the system early in the design phase when user-requested modifications of the design are easier to accomplish and less expensive.

Until recently, there has been a lack of software development tools to expedite the development of realistic user-system interfaces that can be critiqued and examined during system definition and preliminary design. System design tools for user-system interfaces need to produce design expressions or documentation of sufficient detail and scope to permit government reviewers to fully understand, critique, and see the operational consequences of a user-system interface design. Useful tools should have the capability to

analyze or track the problems a user experiences when interacting with a prototype system. Map data retrieval and display capabilities are necessary to prototype C³I, I&W, or situation analysis systems. In addition, the hardware and software architectures selected by the system designer may place constraints on the command structures and the data a user sees and manipulates, the logical relationships between commands, the resulting displays, and the responsiveness of the system to user requests. For the system to perform as required, the system design must result in an environment which is easy to use. Thus, user-system interface prototypes developed without adequate tools are often clumsy. Yet, a good user-system interface is critical for full and easy use of system functions.

Deficiencies in user-system interface design tools are particularly marked in systems employing high-resolution color graphics displays and menus, where text and traditional pen and ink expressions of the design cannot communicate the flow of the interaction, the "feel" of the interactive devices, nor the visual impact of the use of color. Also, these systems do not produce the data required to analyze the user's problems in interacting with the system. A reviewer trying to evaluate whether an interface represents a good design must read a technical description of the menu hierarchy and the meaning and consequences of each command and data entry field. Even with a comprehensive verbal description of the interface, the flow of the interaction is difficult to grasp and the ergonomic issues of color and interactive device utilization are difficult, if not impossible, to communicate. Furthermore, a technical description does not allow the human factors analyst to monitor and analyze data produced by the user interacting with the system.

The commercial marketplace offers some display construction tools, ranging in diversity from "paint" programs for microcomputers to very complex image processing capabilities in support of the video production industry. Although these tools address graphic construction, they do not incorporate all the capabilities needed to produce user-system interface prototypes, particularly for C³I, I&W, situation analysis, and other data fusion systems.

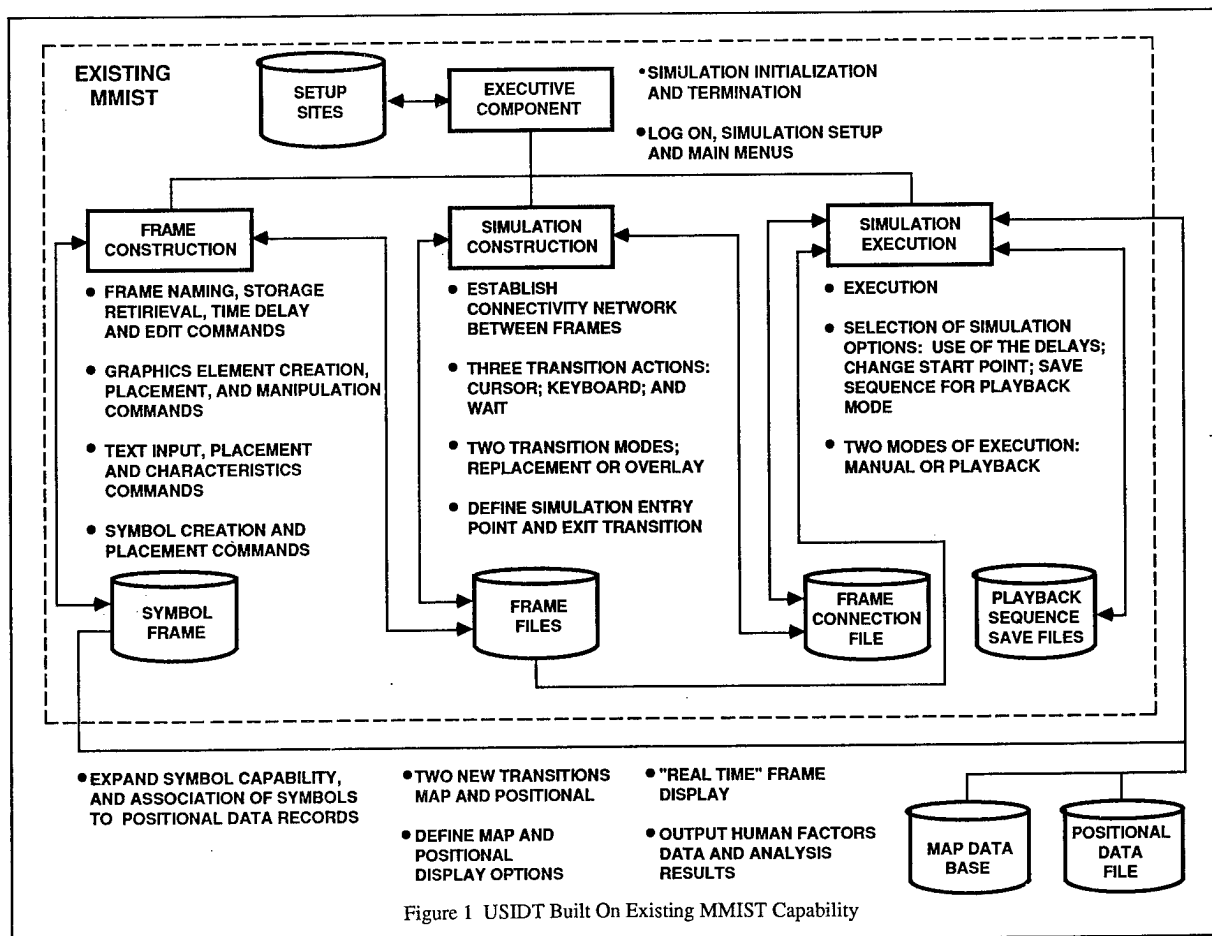


Figure 1 USIDT Built On Existing MMIST Capability

To improve the development productivity on data fusion systems that include interactive graphics displays in their man/machine interfaces, HRB-Singer developed the Man/Machine Interface Simulation Tool (MMIST) that provided the capability to construct prototypical menu and display frames, to connect the developed frames together with user input actions (frame transition actions), and to execute the simulation of the user interaction as a sequence of frame displays interlaced with user actions (Ref. 6). This tool has been used by designers and government design reviewers to develop and evaluate prototype displays, menus, and processing scenarios early in the design phase, and thus ensure the development of user-system interface software packages that are usable and adhere to military standards. A significant productivity gain was achieved as a result of doing a better job of detailing and demonstrating interactive displays/menus and processing sequences early in the design phase (Ref. 6).

This paper presents HRB-Singer's newly developed User-System Interface Design Tool which simultaneously provides the capabilities to build prototypical menu and display frames, define transitions between frames, execute a simulation (with optional global and frame specific timing delays), retrieve and display data from external map and scenario data files as part of the simulation, and monitor user interaction with the simulation. Built upon the existing MMIST software package, USIDT adds a collection capability for user-system interaction data, including the number,

type and location of user errors and analysis of user command linkages, frequencies, and trends in usage. Further, USIDT adds the capability to retrieve and display map and scenario data from external files and improves the flexibility and usability of the previous MMIST software.

Use of USIDT allows early insight into specification of performance requirements and identifies performance problems early in the software development cycle. To the DoD, this means that USIDT will ultimately reduce costs, reduce system maintenance, increase system reliability, and improve the system's usability.

USIDT OVERVIEW

The User-System Interface Design Tool (USIDT) is capable of simulating menu interaction and interactive map and data overlay, including simulation execution time for zooming on the map displays. Further, USIDT provides the capability to collect, analyze and report human factors data not available elsewhere in such a framework. The use of this design tool will improve the user-system interfaces of C³I, I&W, and situation analysis, and will reduce development costs by minimizing late software life cycle rework as result of the improved communication between system designers and government reviewers early in the design phase, when modification of the design is easier to accomplish and less

expensive. Our software development experience has shown that a software function modified at the I&T, or delivery phase of the software life cycle, costs about 70% more to correct than the same modification made during the design phase. Thus, the more precisely a design can be shown to reviewers, such as with the use of the USIDT system, the more likely that problems with the user-system interface design will be identified and corrected early in the software lifecycle, thereby reducing total system development cost and producing the most usable system possible.

Architecture Overview

The Man/Machine Interface Simulation Tool (MMIST) provided the capability to construct prototypical menu and display frames, to connect the developed frames together with user input actions (frame transition actions), and to execute the simulation of the user interaction as a sequence of frame displays interlaced with user actions. USIDT was built upon the MMIST capability and its software architecture continues to reflect MMIST's structure. Figure 1 presents the four major components of the MMIST architecture and the additional capabilities incorporated to produce USIDT.

The Executive Component performs simulation initialization and termination tasks in addition to Logon, Simulation Setup, and Main Menu interaction.

Frame Construction provides the capabilities with which the simulation developer builds individual simulation frames including:

- naming, storing, and retrieving for editing, deleting, and clearing of frames
- graphics commands for placing lines, circles, rectangles, ellipses, polygons, and symbols in frames
- symbol construction commands
- text size, font, orientation, input, and placement commands
- color assignment commands for text, lines, closed figure interiors, and backgrounds
- utility commands to remove pictorial elements from frames, copy and/or move elements, reassign colors, fill closed figure interiors, display reference frames, display user-specified reference grid (and constrain graphic elements to the reference grid), and to redraw the frame at user request
- set frame-specified time delays

Simulation Construction provides the capabilities used by the simulation developer to build a database of user actions (transition actions) which causes the simulation to advance from frame to frame. These capabilities include commands for:

- making, editing, or breaking linkages between simulation frames
- specifying the type (cursor placement, keyboard entry or wait time) and mode (replacement of first frame by second or overlay of second on first) of user action
- defining the starting frame of the simulation and for defining transition actions which cause the simulation to terminate
- displaying and outputting the characteristics of links between frames

Finally, Simulation Execution provides the capabilities to run the developed simulation and to select various execution options, including:

- use of global time delay values and/or the frame specific time delays assigned to each frame in Frame Construction
- selection of a different starting frame than the one specified in Simulation Construction
- requests to save the interactive session to be run for playback at a later time
- requests to play back a previously saved interactive session and the selection of the playback execution mode

Added Capabilities of USIDT

To develop USIDT, we added capabilities to our MMIST tool in the following three major areas: Human Factors, Map and Scenario Data Display, and USIDT User-System Interface Enhancements.

- o Human Factors
 - Added the capability to record, list, and correlate human factors data (user errors and command utilization and linkages)
 - Upgraded the Simulation Execution capability to allow the selection of human factors data collection and reporting
- o Map and Scenario Data Display
 - Added the capability to retrieve and display data from external map and scenario data files
 - Added the capability to connect or track multiple scenario file data records as occurrences of the same unit over time during simulation execution
 - Upgraded the Simulation Setup capability to allow users to:
 - specify the location and extent of the map and scenario data overlay window on the screen
 - specify which map features are to be displayed and the color assigned to the displayed features
 - select the type of symbology to be displayed with scenario data records (point, ellipse, track, and/or symbol)
 - Expanded the Symbol Construction functionality by adding a capability to associate symbols with records in the external scenario file
 - Upgraded the Simulation Construction capability to provide for new user transition actions allowing users to make transitions to map and scenario data displays
- o User System Interface Enhancements
 - Rehosted the VAX 11/780 FORTRAN source code to the Silicon Graphics 3120 workstation
 - Upgraded the Simulation Construction capability to provide for new user transition actions allowing users to select on screen objects as a basis for transition in addition to keyboard input, cursor selection of screen rectangular areas, and time delays
 - Enhanced the Symbol Construction capability by increasing the number of symbols simultaneously available, and improving the symbol storage strategy making it possible to use symbols developed for one simulation in subsequent simulations

Human Factors Report for: DIM

Sun Aug 30 16:15:13 1987

Origin Frame	Destination Frame	Time Of Response	Waiting Time	Transition Number	Transition Type	Transition Description
Start	1	16:14:57	00:00:00	----	Begin	Simulation Initiation
1	Error	16:15:02	00:00:04	----	Text	GOTO0
1	2	16:15:04	00:00:02	2	Text	GOTO2
2	Error	16:15:06	00:00:02	----	Region	(0.334, 0.342)
2	Abnorm Term	16:15:08	00:00:02	----	Region	(0.750, 1.000)

Total Transition Request Errors: 2

Possible Typographical Errors

String Entered	Possible Match
GOTO0	GOTO2

End Of Human Factors Report Generation.

Tue Sep 1 15:10:43 1987

Origin Frame	Destination Frame	Time Of Response	Waiting Time	Transition Number	Transition Type	Transition Description
Start	1	15:09:17	00:00:00	----	Begin	Simulation Initiation
1	2	15:09:20	00:00:03	2	Text	GOTO2
2	3	15:09:21	00:00:00	3	Region	(0.000, 0.651) (0.488, 0.000
3	4	15:09:26	00:00:00	4	Time	Time Delay Transition
4	1	15:09:32	00:00:04	5	Text	GOTO3
1	2	15:09:35	00:00:02	2	Text	GOTO2
2	3	15:09:36	00:00:00	3	Region	(0.000, 0.651) (0.488, 0.000
3	4	15:09:41	00:00:00	4	Time	Time Delay Transition
4	1	15:09:47	00:00:04	5	Text	GOTO3
1	2	15:09:49	00:00:02	2	Text	GOTO2
2	3	15:09:51	00:00:01	3	Region	(0.000, 0.651) (0.488, 0.000
3	4	15:09:55	00:00:00	4	Time	Time Delay Transition
4	1	15:10:01	00:00:04	5	Text	GOTO3
1	2	15:10:04	00:00:03	2	Text	GOTO2
2	3	15:10:06	00:00:02	3	Region	(0.000, 0.651) (0.488, 0.000
3	4	15:10:10	00:00:00	4	Time	Time Delay Transition
4	1	15:10:14	00:00:02	5	Text	GOTO3
1	2	15:10:18	00:00:04	2	Text	GOTO2
2	3	15:10:20	00:00:02	3	Region	(0.000, 0.651) (0.488, 0.000
3	4	15:10:24	00:00:00	4	Time	Time Delay Transition
4	1	15:10:28	00:00:02	5	Text	GOTO3
1	2	15:10:32	00:00:04	2	Text	GOTO2
2	3	15:10:34	00:00:01	3	Region	(0.000, 0.651) (0.488, 0.000
3	4	15:10:38	00:00:00	4	Time	Time Delay Transition
4	Norm Term	15:10:42	00:00:02	6	Text	STOP

Transition Number Sequence Occurrence Rates

Occurrences Sequence

5	2	3	4	5
5	5	2		
1	2	3		
1	3	4		
1	4	6		

Total Transition Request Errors: 0

End Of Human Factors Report Generation.

Figure 2 Sample Human Factors Report

- Upgraded the Simulation Setup capability to allow users to interactively modify the color palette

Reporting Human Factors Data

The Simulation Execution component now records and reports human factors data. This allows a human factors analyst to analyze a proposed design and suggest changes based on data collected during system prototyping. In MMIST, the transition between any pair of simulation frames is controlled by the user performing a transition action, such as typing at a keyboard or positioning a cursor on screen and pressing a button. The simulation previously had a database of all permissible transitions between frames. The existence of this data made it possible for USIDT to record the number of times a user tries to perform an undefined transition (e.g., enters an unknown character string at the keyboard or fails to place the on-screen cursor over the defined rectangular areas or over objects for which transitions exist). The record of unsuccessful transition attempts is summarized and characterized to identify the frequency of typographical errors, and attempts to perform undefined transitions. This record may then be used to identify those transitions which perhaps should be included and that the prototype user expects to be able to perform.

In addition to the simulation log generation and reporting, we installed new software to analyze the transitions log to identify command linkages and the frequency of occurrence of the repetition of set of transitions. Analysis of the frequency of occurrence observations is used to identify command structures which may be awkward and need to be restructured. This analysis can reveal that some commands are being repeated many times and need to be examined for possible simplification, streamlining, or restructuring.

Figure 2 presents two samples of the human factors report. The first section of the report is a log of the transitions performed containing data on the frame from which the transition occurred, the frame to which the simulation advanced, the time of day, the time required for the user to perform the transition, the record number of the transition in the permissible transitions data base, the type of transition attempted, and a transition action description, either in the form of a text string, normalized screen coordinates, object Pick Id, or user-specified time delay. Following the simulation log, the report displays the list of any multiple occurrences of the transition series sets. The total number of transition errors made during the simulation execution follows the list of transition sequence repetitions. If any typographical errors have been made, then the report contains a section identifying the character strings in error and the possible matches in the permissible transitions database.

Displaying Map and Scenario File Data

The prototyping of systems utilizing interactive map displays (C³I, I&W, and Situation Analysis) require zooming and panning on user-selected map centers and at user-selected map scales. User-system interface simulation systems which do not provide for zooming and panning of map displays can not adequately simulate this functionality in other ways. Further, since the display of data against the map is the critical element, prototyping tools for these types of simulations need a mechanism for incorporating scenario data. To address these deficiencies in our MMIST tool (and in the products available in the commercial

marketplace), we have incorporated map and scenario data base retrieval and display capabilities. USIDT simulations are able to zoom and pan on map displays and scenario data overlays during simulation execution, and thereby provide the most realistic prototype of the system possible.

To provide for the simulation of interactive map and scenario data display, we installed external map database and scenario file data retrieval and display capabilities. To provide for the realistic treatment of scenario data display, we installed a capability to associate user-developed symbols with scenario data for the purpose of displaying symbols during simulation execution. In addition to symbols, scenario data may be optionally displayed with ellipses of locational uncertainty, geolocation indicators, identification code character strings, and track or patrol paths for moving units. Since scenario data is often dynamic with respect to time, we installed the capability to place/move/remove scenario data display symbology on/over/from the map background as the scenario data file indicates the movement through time of mobile units.

The simulation developer creates the external scenario data file outside the USIDT tool using the host computer's text editor. A scenario file data record includes:

- An identification code for the unit (up to 5 characters)
- Unit mobility code (fixed, hostile track, friendly patrol path)
- Time of first and last occurrence of the unit
- For mobile units, the location, time, and locational uncertainty ellipse parameters for each point in the track or patrol path

Having the scenario data in an external file allows the USIDT user to easily modify simulation scenario data. Thus, new scenarios can be developed, experimented with, and demonstrated without modifying the simulation's internal structure in any way.

The external map database incorporated into USIDT contains 1.5 million points in six categories:

- Shoreline
- Topographic relief contours (8 levels)
- Bathymetric depth contours (8 levels)
- Rivers, lakes, canals and seasonal water bodies
- Political boundaries
- Cities (names, country capitals, and four levels of population size)

This map database was constructed by direct digitization from the Defense Mapping Agency (DMA) map series 1150, "The World." The database exists in four levels of resolution to simultaneously support map detail and display throughput requirements. Series 1150 maps are in Mercator projection, and cover the globe only from 84°N to 70°S. The polar regions of the world were filled in from equivalent scale maps from other sources, most notably the DMA Polar Regions Atlas and National Geographic Maps of the Arctic and Antarctic. Figure 3 presents a sample of the map database incorporated into USIDT. The USIDT user has control over which of the six classes of map features to be used in simulation of a user-system interface, as well as control over the color assigned to each feature and the threshold value of the scale at which the feature first appears.



Figure 3 Sample Map Data Base-Great Britain and Ireland

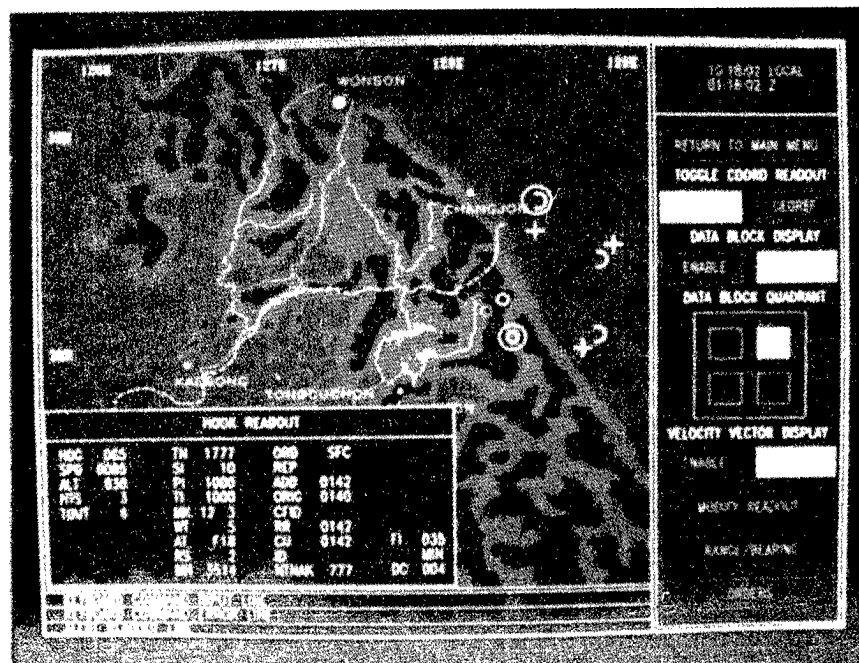


Figure 4 Prototype

C³I, I&W and situation analysis systems utilize map displays in a variety of ways and display maps in a variety of sizes and locations on CRT screens. USIDT provides thirteen combinations of map window sizes and locations. These combinations include:

- Full screen map window
- Four 3/4 screen (in both height and width) map windows
- Four 1/2 screen (in both height and width) map windows
- Four 1/2 by full screen (or full by 1/2) map windows

Further, USIDT provides the simulation developer with capabilities to select map features, colors, fill options for polygon-based features, feature selection based on scale of displayed maps, and the type of symbology to be displayed for each scenario data item. Figure 4 presents a sample of a USIDT map and scenario data display created by the user selecting the following options:

- Upper left corner 3/4 screen map window
- Fill of polygon features
- All map features except bathymetric depth
- Scenario data displayed with symbols and tracks (moving units only)
- User specified map colors

The Simulation Construction component of USIDT now permits the simulation developer to connect the simulation frames to the map and scenario data displays, thereby permitting the user to develop part of the display screen as table displays and interactive menus and part of the display screen for the real-time construction of map and scenario data displays.

The USIDT User-System Interface

In response to requests from users, USIDT has been enhanced in a number of other ways in order to increase speed in retrieving complex frames, interactively define color palettes, increase symbols, and increase user transition selection. This allows the building of more robust simulations and results in not only more timely prototypes, but also more effective evaluation of proposed designs by DoD reviewers at less cost. Perhaps the most important of these additional enhancements was the porting of the software from the Ramtek 9465/VAX 11/780 configuration to a Silicon Graphics workstation. The former configuration suffered from an inadequate graphics engine coupled with an overburdened minicomputer with the result that MMIST was painfully slow at retrieving and displaying complex simulation frames and was slow to react to user commands. The graphics engine of the Silicon Graphics 3120 is at least ten times faster at performing graphics instructions than the Ramtek. The Silicon Graphics has approximately the same computation power as the VAX 11/780 and is dedicated to the USIDT user rather than shared by many VAX users.

Other user-requested enhancements included a capability to interactively redefine the color palette available to the simulation frame developer, and to provide for the development and use of more symbols. MMIST allowed the user only 14 simultaneous symbols. These symbols were stored as a group and were very difficult to reuse in subsequent simulations. USIDT now permits the user to have 99 simultaneously available symbols. Each symbol is stored as a discrete unit and much more transportable. USIDT provides interactive color palette modification allowing the simulation developer to modify any or all of the colors in the 78-color palette. For example, if a simulation under development

requires 10 grey scales (the USIDT default color palette contains only 6 grey scales), the simulation developer may now elect to reassign any four other colors to the required grey values.

To further improve the flexibility of the simulation of the user-system interface, an additional type of user transition action has been added to three previously existing transition formats (keyboard input, occurrence of a cursor button push in rectangular subregions of the screen, and wait times). This new user transition action is cursor selection (by Pick Id) of objects displayed on the screen. Often users select commands via on-screen menus. For this type of user action, the rectangular area selection by cursor is appropriate. Perhaps nearly as often, users point to or select an object for an additional data display. For this type of action, an object-oriented transition is more appropriate than the rectangular area transition.

SUMMARY

Rapid prototyping of user-system interfaces helps solidify system requirements and identify many of the defective or operationally awkward portions of a design which would otherwise go undetected until installation of the completed system. Prototyping the user-system interface allows government reviewers to examine the proposed system design, to suggest changes to the structure or flow of the interaction, and to observe the results of the modifications. The more precisely a design can be shown to reviewers, such as with the use of our USIDT system, the more likely that problems with the user-system interface design will be identified and corrected early in the software life cycle, thereby reducing total system development cost and producing the most usable system possible. Further, prototyping makes it possible for system designers to collect and analyze data on the user's interaction with the prototype expression of the design. The prototyping of the C³I, I&W, and situation analysis systems utilizing interactive menu, map, and scenario data overlays, need to be able to pan and zoom on map and data displays, and this is only possible when map and data files are accessible to the simulation. HRB-Singer's USIDT is the most complete user-system interface prototyping tool available today.

REFERENCES

1. J. Lohr, The Tactical Man-Machine Interface, Signal (Oct. 1980).
2. B. Dickinson, "Developing Structured Systems - A Methodology Using Structural Techniques," Yourdon Press (1981).
3. R. L. Glass, Recommended: A Minimum Standard Software Toolset, Software Engineering Notes, 7:4 (Oct. 1982).
4. S. Smith, "User-System Interface Design for Computer-Based Information Systems," Mitre Corporation Report No. ESD-TR-082-132 (April 1982).
5. J. Foley, et al., The Human Factors of Computer Graphics Interaction Techniques, IEEE Computer Graphics and Applications, 4:11 (Nov. 1984).
6. S. Morgan, J. Kindon, and A. Nauda, "A Man-Machine Interface Simulator," Proc. 1985 Winter Simulation Conference (Dec. 1985).

AUTHORS



STEPHEN L. MORGAN is a Principal Engineer in the Tactical Processing and Reporting Section at HRB-Singer. He has 15 years of managerial and working experience in the design and development of software systems for the processing of remotely sensed data, with emphasis on the development of user-system interfaces using advanced computer graphics

technologies and geographic information systems. His responsibilities include the management of software system specifications, design, development, and integration tasks with particular emphasis on software architecture design, complex algorithm development, and development of user-friendly user-system interfaces. He has written several technical papers related to computer graphics technology and user-system interface simulation. He is currently serving as chairman of HRB-Singer's Computer Graphics Technology Working Group and the president of the Central Pennsylvania Chapter of the National Computer Graphics Association. Mr. Morgan holds a Geography and Environmental Engineering degree from The Johns Hopkins University and a Master's degree in Geography from The Pennsylvania State University.



ALEXANDER NAUDA is the Staff Technical Advisor to the Vice President of Quality and Strategic Development at HRB-Singer, where he supports strategic business development in the areas of digital signal processing, statistical data analysis, VLSI technology, and the selection and technical direction of research and development projects in signal acquisition,

analysis, and processing. His 20 years of academic and industrial experience encompass algorithm development, mathematical modeling, computer simulation, and performance analysis/evaluation of signal processing and communication systems. A registered professional engineer, he has consulted for Singer, NASA, Alcoa, PPG Industries, and the Technology Training Corporation. A senior member of IEEE, he recently served as chairman of the Central Pennsylvania Chapter of the IEEE Acoustics, Speech, and Signal Processing Society. Dr. Nauda received his B. S., M. S., and Ph. D. degrees in Electrical Engineering from the University of Pittsburgh.

THIS PAGE LEFT BLANK INTENTIONALLY

TRACK FILE FUSION

**Jeff Brandstadt
GE Aerospace
Aerospace Electronic Systems
Utica, New York 13503**

ABSTRACT

In general, the most accurate method of data fusion (in the Minimum Mean-Squared Error (MMSE) sense) is to receive uncorrelated, linear data and fuse it with a Minimum Mean-Squared Error filter. In practice, however, the data rate and the volume of measurement data available in a multispectral sensor system are so high that operating a MMSE filter on every available data sample is an unreasonable solution to the data fusion problem. Instead, it may be more feasible for a fusion processor to accept filtered data and to generate a suboptimal estimate. In this paper, we develop and analyze two methods to fuse filtered data. Each method operates on the track files generated by individual sensor subsystems. The first method accepts track files and their error covariances; then assuming that the track files from different sensors are uncorrelated and that every sensor supplies a track file at every update interval, the algorithm calculates the MMSE fused estimate for each target. The second method is an extension of the first; it accounts for the cross-correlation between different sensor estimates.

The two track file fusion methods developed in this paper have been published previously by other authors. We have derived them again for completeness and run simulations to analyze their performance. In this paper, we present comparisons of the performance of the track file techniques. We developed an ideal simulation that generates individual position-velocity track files from independent Cartesian position measurements. The simulation fuses the track files with each of the techniques discussed above, and we present the final tracking errors on a single plot for comparison.

TRACK FILE FUSION

Jeff Brandstadt
GE Aerospace
Aerospace Electronic Systems
Utica, New York 13503

1.0 INTRODUCTION

The following paper originated from a very specific question about a very specific problem. While attempting to fuse filtered estimates from two different sources, we questioned the significance of the cross-correlation term in the computation of the fusion gain. In the process of determining the cross-correlation term's significance, we raised a number of other questions about the fusion of two track estimates and about fusion in general. As a result, we wrote the following paper for our own reference. It has become a tutorial that discusses a few fusion techniques for kinematic parameters. Specifically, we discuss the significance of the cross-correlation terms in the gain calculation. This paper is not a discussion of the accuracy improvement provided by data fusion. In our experience, when two sensing systems measure different target parameters (e.g., a radar system's range/range rate data combined with an infrared system's azimuth/elevation data), data fusion offers unquestionable improvement in track accuracy. For a more detailed discussion, we reference the reader to R. Yannone's paper on sensor fusion [Ref. 4].

2.0 FUSION

Fusion is such a general topic that everyone's definition and application is a little different. Before we begin any discussion of fusion, it is best to give our definition of the term. GE Aerospace Electronic Systems (AES), located in Utica, New York, designs airborne sensors (Infrared Search and Track Systems (IRSTS), fighter and surveillance radar, electronic countermeasures (ECM), and electronic counter-countermeasures (ECCM) receivers); therefore, all of our fusion applications apply to airborne detection, tracking, and identification problems. Inputs to the fusion process could be raw kinematic measurements consisting of range,

range rate, azimuth, and elevation measurements; or attribute measurements like JEM-line discriminations, emitter frequencies, infrared signatures, etc.; or the data could be filtered data from both on-board and off-board assets. Figure 1 illustrates the origins of the airborne data.

We define the fusion processor as the function that accepts data from a variety of sources and combines them to produce one composite track file on each target. The composite track file has better track accuracy and higher confidence identification on each target than any one sensor system. In most systems, the composite track file is passed on to an information processor that assesses the overall battle situation based on the composite track files and a priori mission database, and from this assessment, recommends a response or allocates the system resources in an optimal manner. The following discussion applies to one small part of the fusion processor. It does not attempt to discuss possible situation assessment or response techniques.

We break the fusion processor itself into three subfunctions:

- Association/Correlation.
- Kinematic fusion.
- Attribute fusion.

Figure 2 illustrates these subfunctions.

The Association/Correlation subfunction accepts the incoming measurement or filtered estimate and tags it to the appropriate composite track file or files. (If there is a large uncertainty as to which target generated the measurement, then a probabilistic association of the measurement to multiple track files is probably the best association technique.) Once we have correlated the measurement to a composite track file, the Kinematic Fusion subfunction updates the track file kinematics with the

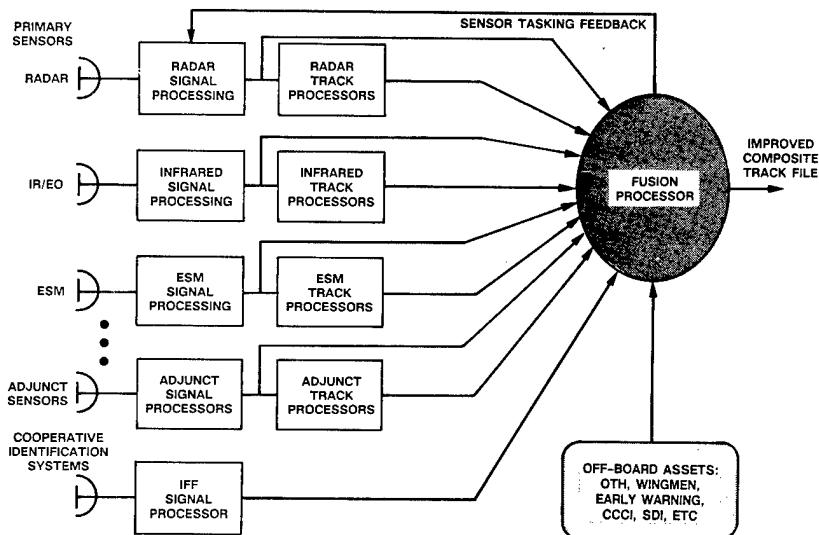


Figure 1. Airborne Sensor Fusion Problem

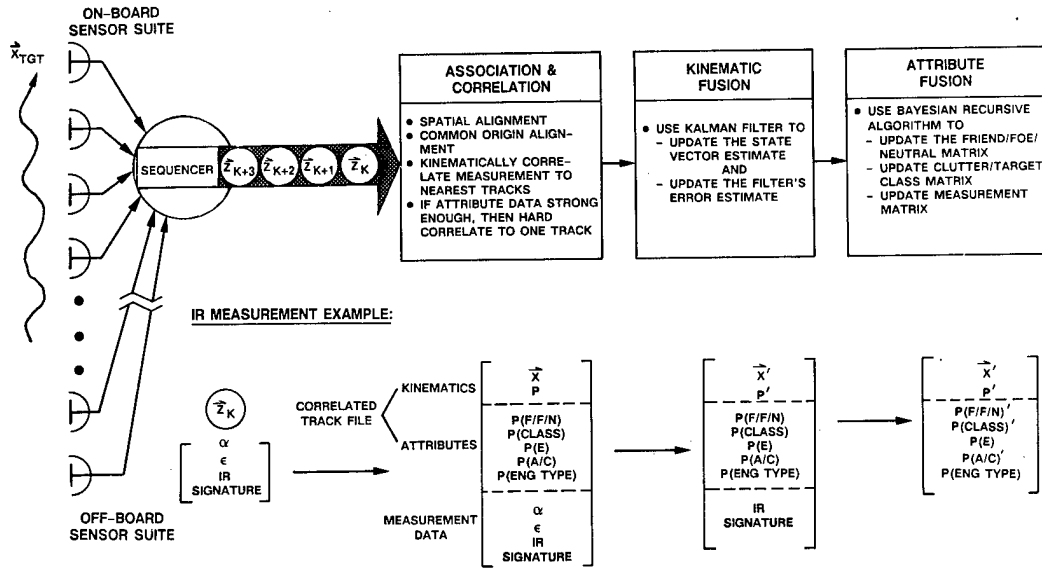


Figure 2. Top-Level Fusion Definition

kinematic measurement parameters via a Kalman filter, and the Attribute Fusion subfunction employs a probabilistic or expert system technique to update the track file's attribute vector. The lower portion of Figure 2 illustrates how each of these subfunctions affects the composite track file assuming the measurement originates from an IR sensor. The Association/Correlation subfunction appends the measurement data to the composite track file. The Kinematic Fusion function updates the file's kinematic state vector estimate ($\hat{\mathbf{x}}$) and associated error covariance matrix (\mathbf{P}) with the azimuth (α) and elevation (ϵ) measurement. The Attribute Fusion subfunction updates the track file's Friend/Foe/Neutral (F/F/N), class matrix (CLASS) and aircraft platform identification (A/C) with the new IR signature information. In the following discussion, we will concentrate on the Kinematic Fusion subfunction. We will assume that the Association/Correlation subfunction performed perfectly and tagged the measurement to the correct composite track file.

There are really two types of kinematic fusion: measurement fusion and track file fusion. The difference is in the type of input. If the incoming data is uncorrelated in time and from sensor to sensor, then we employ the measurement fusion technique. If, however, the data is filtered by the sensor and therefore is definitely time-correlated and perhaps even sensor-to-sensor-correlated, then we use the track file fusion technique. The major points that we discuss in this paper deal with track file fusion, but because of its similarity to measurement fusion, we discuss measurement fusion as an introduction to track file fusion.

2.1 Measurement Fusion

The measurement fusion algorithm accepts raw uncorrelated measurements from the sensor suite and uses them to update the composite track file kinematics. Assuming that we require a linear update, we can express the updated estimate as a linear combination of the previous estimate and the incoming measurement vector,

$$\hat{\mathbf{x}}(k/k) = \mathbf{K}_1 \cdot \hat{\mathbf{x}}(k/k-1) + \mathbf{K}_2 \cdot \hat{\mathbf{z}}_i(k), \quad (1)$$

where $\hat{\mathbf{x}}(k/k-1)$ is the $n \times 1$ kinematic state vector estimate valid at t_k given data up to time t_{k-1} , $\hat{\mathbf{z}}_i(k)$ is the $m_i \times 1$ measurement vector generated by the i th sensor, and \mathbf{K}_1 and \mathbf{K}_2 are the $n \times n$ and $n \times m_i$

linear gains. Our objective is to determine \mathbf{K}_1 and \mathbf{K}_2 so that $\hat{\mathbf{x}}(k/k)$ is an unbiased, minimum mean-square error estimate of the true kinematic state vector, $\mathbf{x}(k)$.

We begin by looking at the error terms associated with equation 1. Assume that the measurement contains additive, white noise, then

$$\hat{\mathbf{z}}_i(k) = \mathbf{C}_i \cdot \mathbf{x}(k) + \hat{\mathbf{v}}_i(k), \quad (2)$$

where \mathbf{C}_i is a $m_i \times n$, linear transformation from state-space to measurement-space, and $\hat{\mathbf{v}}_i(k)$ is the additive, zero-mean, white noise vector.

The two estimates in equation 1 may be expressed in terms of their corresponding errors,

$$\hat{\mathbf{x}}(k/k-1) = \mathbf{x}(k) + \hat{\mathbf{e}}(k/k-1), \quad (3)$$

and

$$\hat{\mathbf{x}}(k/k) = \mathbf{x}(k) + \hat{\mathbf{e}}(k/k). \quad (4)$$

Substituting these expressions for the estimates and the expression for the measurement vector into equation 1, we get

$$\begin{aligned} \hat{\mathbf{x}}(k) + \hat{\mathbf{e}}(k/k) &= \mathbf{K}_1 [\hat{\mathbf{x}}(k) + \hat{\mathbf{e}}(k/k-1)] + \\ &\quad \mathbf{K}_2 \cdot [\mathbf{C}_i \cdot \hat{\mathbf{x}}(k) + \hat{\mathbf{v}}_i(k)], \end{aligned} \quad (5)$$

or the error at t_k given the data at t_k is

$$\hat{\mathbf{e}}(k/k) = (\mathbf{K}_1 + \mathbf{K}_2 \mathbf{C}_i - \mathbf{I}) \hat{\mathbf{x}}(k) + \mathbf{K}_1 \hat{\mathbf{e}}(k/k-1) + \mathbf{K}_2 \cdot \hat{\mathbf{v}}_i(k). \quad (6)$$

If we require an unbiased estimate, and we do, then

$$\mathbf{K}_1 + \mathbf{K}_2 \mathbf{C}_i - \mathbf{I} = [0], \quad (7)$$

or

$$\mathbf{K}_1 = \mathbf{I} - \mathbf{K}_2 \mathbf{C}_i. \quad (8)$$

For convenience let

$$\mathbf{K}_2 = \mathbf{K}, \quad (9)$$

then

$$\mathbf{K}_1 = \mathbf{I} - \mathbf{K} \mathbf{C}_i \quad (10)$$

Now substitute these expressions for \mathbf{K}_1 and \mathbf{K}_2 into equation 6 and find that the unbiased error is

$$\hat{\mathbf{e}}(k/k) = (\mathbf{I} - \mathbf{K} \mathbf{C}_i) \cdot \hat{\mathbf{e}}(k/k-1) + \mathbf{K} \cdot \hat{\mathbf{v}}_i(k) \quad (11)$$

We have to solve for the gain, \mathbf{K} , that minimizes the error vector $\hat{\mathbf{e}}(k/k)$ at every time t_k .

We minimize the total length of the error vector, $\hat{\mathbf{e}}(k/k)$, by minimizing the sum of the squares of its individual components. Because $\hat{\mathbf{e}}(k/k)$ is stochastic, we choose to minimize the expected value of the squared errors. Thus, set the gradient of $E[\hat{\mathbf{e}}^T(k/k) \hat{\mathbf{e}}(k/k)]$ with respect to \mathbf{K} equal to zero, then solve for \mathbf{K} .

$$\nabla_{\mathbf{K}} \{E[\hat{\mathbf{e}}^T \hat{\mathbf{e}}]\} = [0], \quad (12)$$

where we let $\hat{\mathbf{e}} = \hat{\mathbf{e}}(k/k)$ for convenience, and $\nabla_{\mathbf{K}}$ is the matrix gradient operator,

$$\nabla_K = \begin{bmatrix} \frac{\partial}{\partial k_{11}} & \frac{\partial}{\partial k_{12}} & \cdots & \frac{\partial}{\partial k_{1m}} \\ \frac{\partial}{\partial k_{21}} & \frac{\partial}{\partial k_{22}} & \cdots & \frac{\partial}{\partial k_{2m}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial}{\partial k_{n1}} & \frac{\partial}{\partial k_{n2}} & \cdots & \frac{\partial}{\partial k_{nm}} \end{bmatrix} \quad (13)$$

with properties

$$\nabla_K (\vec{V}^T K \vec{W}) = \vec{V}^T \vec{W}^T \quad (14)$$

and

$$\nabla_K (\vec{V}^T K^T K \vec{W}) = K (\vec{W}^T \vec{V}^T + \vec{V}^T \vec{W}^T) \quad (15)$$

From the previous expression for $\epsilon(k/k)$, equation 11, we can write

$$\vec{\epsilon} = \vec{\epsilon}(k/k-1) + K \cdot [\vec{v}_i(k) - C_i \cdot \vec{\epsilon}(k/k-1)] \quad (16)$$

thus, the equation we must solve for K is

$$\nabla_K \left[E \left\{ \vec{\epsilon}^T(k/k-1) + [\vec{v}_i^T(k) - \vec{\epsilon}^T(k/k-1) C_i^T] K^T \right\} \right. \\ \left. \cdot \left\{ \vec{\epsilon}(k/k-1) + K [\vec{v}_i(k) - C_i \cdot \vec{\epsilon}(k/k-1)] \right\} \right] = [0] \quad (17)$$

or

$$\nabla_K \left(E \left\{ \vec{\epsilon}^T(k/k-1) \vec{\epsilon}(k/k-1) + \vec{\epsilon}^T(k/k-1) K [\vec{v}_i(k) - C_i \cdot \vec{\epsilon}(k/k-1)] \right. \right. \\ \left. \left. + [\vec{v}_i^T(k) - \vec{\epsilon}^T(k/k-1) C_i^T] K^T \vec{\epsilon}(k/k-1) \right. \right. \\ \left. \left. + [\vec{v}_i^T(k) - \vec{\epsilon}^T(k/k-1) C_i^T] K^T K [\vec{v}_i(k) - C_i \cdot \vec{\epsilon}(k/k-1)] \right\} \right) = [0] \quad (18)$$

Employing the properties listed in equations 14 and 15, and noting that

$$\nabla_K \{ E[\vec{\epsilon}^T(k/k-1) \vec{\epsilon}(k/k-1)] \} = [0], \quad (19)$$

allows us to perform the gradient operation in equation 18 and get

$$2E \{ \vec{\epsilon}(k/k-1) \cdot [\vec{v}_i^T(k) - \vec{\epsilon}^T(k/k-1) C_i^T] \\ + K \cdot [\vec{v}_i(k) - C_i \cdot \vec{\epsilon}(k/k-1)] \cdot [\vec{v}_i^T(k) - \vec{\epsilon}^T(k/k-1) C_i^T] \} = [0] \quad (20)$$

where we have made use of the fact that the products in equation 18 are scalars so that

$$\vec{\epsilon}^T(k/k-1) K [\vec{v}_i(k) - C_i \cdot \vec{\epsilon}(k/k-1)] \\ = [\vec{v}_i^T(k) - \vec{\epsilon}^T(k/k-1) C_i^T] K^T \vec{\epsilon}(k/k-1) \quad (21)$$

Multiplying the terms out in equation 20, we get

$$E \{ \vec{\epsilon}(k/k-1) \cdot \vec{v}_i^T(k) \} - E \{ \vec{\epsilon}(k/k-1) \cdot \vec{\epsilon}^T(k/k-1) \} C_i^T \\ + K E \{ \vec{v}_i(k) \vec{v}_i^T(k) \} - K E \{ \vec{v}_i(k) \vec{\epsilon}^T(k/k-1) \} C_i^T \\ - K C_i E \{ \vec{\epsilon}(k/k-1) \vec{v}_i^T(k) \} + K C_i E \{ \vec{\epsilon}(k/k-1) \vec{\epsilon}^T(k/k-1) \} C_i^T \\ = [0] \quad (22)$$

Recall the assumption that the measurement noise is zero-mean and white. Because of this assumption, the measurement noise at t_k , $\vec{v}_i(k)$, must be uncorrelated with the estimate error based on measurements occurring through t_{k-1} , $\vec{\epsilon}(k/k-1)$. Therefore

$$E \{ \vec{\epsilon}(k/k-1) \cdot \vec{v}_i^T(k) \} = [0], \quad (23)$$

and

$$E \{ \vec{v}_i(k) \cdot \vec{\epsilon}^T(k/k-1) \} = [0], \quad (24)$$

and the equation we must solve for K becomes

$$K \{ C_i E \{ \vec{\epsilon}(k/k-1) \vec{\epsilon}^T(k/k-1) \} C_i^T + E \{ \vec{v}_i(k) \vec{v}_i^T(k) \} \} \\ = E \{ \vec{\epsilon}(k/k-1) \vec{\epsilon}^T(k/k-1) \} C_i^T. \quad (25)$$

Let the error covariance matrix be

$$P(k/k-1) = E \{ \vec{\epsilon}(k/k-1) \vec{\epsilon}^T(k/k-1) \} \quad (26)$$

and the measurement noise covariance matrix be

$$R_i(k) = E \{ \vec{v}_i(k) \vec{v}_i^T(k) \}, \quad (27)$$

then equation 25 becomes

$$K \{ C_i \cdot P(k/k-1) \cdot C_i^T + R_i(k) \} = P(k/k-1) \cdot C_i^T \quad (28)$$

Solving for K, we find our optimal measurement fusion gain is

$$K = P(k/k-1) \cdot C_i^T \cdot [C_i \cdot P(k/k-1) \cdot C_i^T + R_i(k)]^{-1}, \quad (29)$$

where C_i and $R_i(k)$ depend on the characteristics of the i th sensor.

Now that we know what K is, we can use it to fuse the incoming sensor measurement $\vec{z}_i(k)$ with the correlated kinematic state vector estimate $\hat{\vec{X}}(k/k-1)$. Applying the relationships we found for K_1 and K_2 to the expression for the fused track file $\hat{\vec{X}}(k/k)$ (equation 1), we have

$$\hat{\vec{X}}(k/k) = \hat{\vec{X}}(k/k-1) + K \cdot [\vec{z}_i(k) - C_i \cdot \hat{\vec{X}}(k/k-1)] \quad (30)$$

This, combined with equation 29, makes up our measurement fusion algorithm. The only problem is that we have the unknowns $\hat{\vec{X}}(k/k-1)$ and $P(k/k-1)$.

The first unknown is the predicted state vector estimate given data from t_{k-1} . If the true state vector equation is

$$\vec{X}(k) = \Phi \cdot \vec{X}(k-1) + B \cdot \vec{u}(k-1) + G \cdot \vec{w}(k-1) \quad (31)$$

where Φ is the state transition matrix from t_{k-1} to t_k , $B \cdot \vec{u}(k-1)$ is the deterministic forcing function, and $G \cdot \vec{w}(k-1)$ is zero-mean, white noise process noise; then the best prediction of the state estimate at time t_{k-1} to time t_k is

$$\hat{\vec{X}}(k/k-1) = \Phi \cdot \hat{\vec{X}}(k-1/k-1) + B \cdot \vec{u}(k-1). \quad (32)$$

The error associated with this prediction is

$$\vec{\epsilon}(k/k-1) = \Phi \cdot \vec{\epsilon}(k-1/k-1) - G \cdot \vec{w}(k-1), \quad (33)$$

thus, the error covariance matrix is

$$P(k/k-1) = E \{ [\Phi \vec{\epsilon}(k-1/k-1) - G \vec{w}(k-1)] [\vec{\epsilon}^T(k-1/k-1) \Phi^T - \vec{w}^T(k-1) G^T] \} \quad (34)$$

or

$$P(k/k-1) = \Phi \cdot E \{ \vec{\epsilon}(k-1/k-1) \vec{\epsilon}^T(k-1/k-1) \} \cdot \Phi^T \\ - G \cdot E \{ \vec{w}(k) \vec{\epsilon}^T(k-1/k-1) \} \cdot \Phi^T \\ - \Phi \cdot E \{ \vec{\epsilon}(k-1/k-1) \vec{w}^T(k-1) \} \cdot G^T \\ + G \cdot E \{ \vec{w}(k-1) \vec{w}^T(k-1) \} \cdot G^T \quad (35)$$

The cross-terms in equation 35 equal zero because the error contains no data from time t_k , and data from time t_{k-1} contains $\vec{w}(k-2)$, not $\vec{w}(k-1)$. Therefore, $\vec{w}(k-1)$ and $\vec{\epsilon}(k-1/k-1)$ are uncorrelated. If we let

$$P(k-1/k-1) = E \{ \vec{\epsilon}(k-1/k-1) \vec{\epsilon}^T(k-1/k-1) \} \quad (36)$$

and the process noise covariance matrix be

$$Q(k) = E[\tilde{w}(k-1) \tilde{w}^T(k-1)], \quad (37)$$

then the predicted state estimate's error covariance matrix becomes

$$P(k/k-1) = \Phi \cdot P(k-1/k-1) \cdot \Phi^T + G \cdot Q(k-1) \cdot G^T \quad (38)$$

where the only remaining unknown is $P(k-1/k-1)$ or the error covariance valid at the same time as the data, $P(k/k)$. We find $P(k/k)$ from

$$P(k/k) = E[\tilde{e}(k/k) \tilde{e}^T(k/k)] \quad (39)$$

Substituting the equation for $\tilde{e}(k/k)$ (equation 11) into the above expression for $P(k/k)$, we find that

$$P(k/k) = E \left\{ [(I - K C_i) \tilde{e}(k/k-1) + K \tilde{v}_i(k)] [\tilde{e}^T(k/k-1) (I - K C_i)^T + \tilde{v}_i^T(k) K^T] \right\} \quad (40)$$

Recalling that the cross-terms' expected values equal zero (equations 23 and 24), we have

$$P(k/k) = (I - K C_i) \cdot E[\tilde{e}(k/k-1) \tilde{e}^T(k/k-1)] \cdot (I - K C_i)^T + K \cdot E[\tilde{v}_i(k) \tilde{v}_i^T(k)] \cdot K^T \quad (41)$$

which simplifies when we recall the expressions for $P(k/k-1)$ and $R_i(k)$ (equation 26 and 27).

$$P(k/k) = (I - K C_i) \cdot P(k/k-1) \cdot (I - K C_i)^T + K R_i(k) K^T \quad (42)$$

After substituting for K in this equation and application of the matrix inversion lemma, we can show that

$$P(k/k) = (I - K C_i) \cdot P(k/k-1) \quad (43)$$

which completes our derivation of the measurement fusion filter.

The entire recursive algorithm is (from equations 32, 38, 29, 43, and 30, respectively)

$$\hat{\tilde{x}}(k/k-1) = \Phi \cdot \hat{\tilde{x}}(k-1/k-1) + B \cdot \tilde{u}(k-1) \quad (44)$$

$$P(k/k-1) = \Phi \cdot P(k/k-1) \cdot \Phi^T + G \cdot Q(k-1) \cdot G^T \quad (45)$$

$$K = P(k/k-1) C_i^T \cdot [C_i P(k/k-1) C_i^T + R_i(k)]^{-1} \quad (46)$$

$$P(k/k) = (I - K C_i) \cdot P(k/k-1) \quad (47)$$

$$\hat{\tilde{x}}(k/k) = \hat{\tilde{x}}(k/k-1) + K \cdot [\tilde{z}_i(k) - C_i \cdot \hat{\tilde{x}}(k/k-1)] \quad (48)$$

Notice that this is the exact form of the Kalman filter except that the state-space/measurement-space transformation C_i , the measurement covariance $R_i(k)$, and of course the data itself $\tilde{z}_i(k)$ all vary as a function of the sensor that generated $\tilde{z}_i(k)$. Figure 3 illustrates the algorithm in block diagram form. Next we show how this form differs from the track file fusion case.

2.2 Track File Fusion

Processing every measurement through the measurement fusion algorithm discussed above, many times due to unavailability of data or processing restrictions, is infeasible for each measurement on every degree. In these cases we must fuse data that have been filtered to some degree. As a first step in developing an algorithm to perform this task, let us examine the two-sensor problem illustrated in Figure 4. Both sensors generate their own estimates of the target's kinematic state vector. Assuming that we have already correlated the estimates, what is the linear, unbiased, minimum mean-squared error estimator that combines the filtered data into one composite track estimate?

The problem is very similar to the measurement fusion problem. We begin by requiring a linear combination of the two sensor estimates.

$$\hat{\tilde{x}}(k) = K_1 \cdot \hat{\tilde{x}}_1(k/n) + K_2 \cdot \hat{\tilde{x}}_2(k/m), \quad (49)$$

where $\hat{\tilde{x}}(k)$ is the composite estimate, $\hat{\tilde{x}}_1(k/n)$ is the first sensor's estimate of $\tilde{x}(k)$ at time t_k given data at time t_n , $\hat{\tilde{x}}_2(k/m)$ is the second sensor's estimate of $\tilde{x}(k)$ at time t_k given data at time t_m , and K_1 and K_2 are the linear gain matrices. As in the measurement fusion case, we find an expression for the error in $\hat{\tilde{x}}(k)$. If we express the errors in the estimates as

$$\tilde{\tilde{x}}(k) = \hat{\tilde{x}}(k) + \tilde{e}(k), \quad (50)$$

$$\hat{\tilde{x}}_1(k/n) = \tilde{\tilde{x}}(k) + \tilde{e}_1(k/n), \quad (51)$$

and

$$\hat{\tilde{x}}_2(k/m) = \tilde{\tilde{x}}(k) + \tilde{e}_2(k/m), \quad (52)$$

where we have decided to estimate the composite state, $\tilde{\tilde{x}}(k)$, in the same space as the first sensor's estimate, $\hat{\tilde{x}}_1(k/n)$, then following the previous measurement fusion derivation

$$\tilde{\tilde{x}}(k) + \tilde{e}(k) = K_1 \cdot [\tilde{\tilde{x}}(k) + \tilde{e}_1(k/n)] + K_2 \cdot [\tilde{\tilde{x}}(k) + \tilde{e}_2(k/m)] \quad (53)$$

or

$$\tilde{e}(k) = (K_1 + K_2 C - I) \cdot \tilde{\tilde{x}}(k) + K_1 \cdot \tilde{e}_1(k/n) + K_2 \cdot \tilde{e}_2(k/m), \quad (54)$$

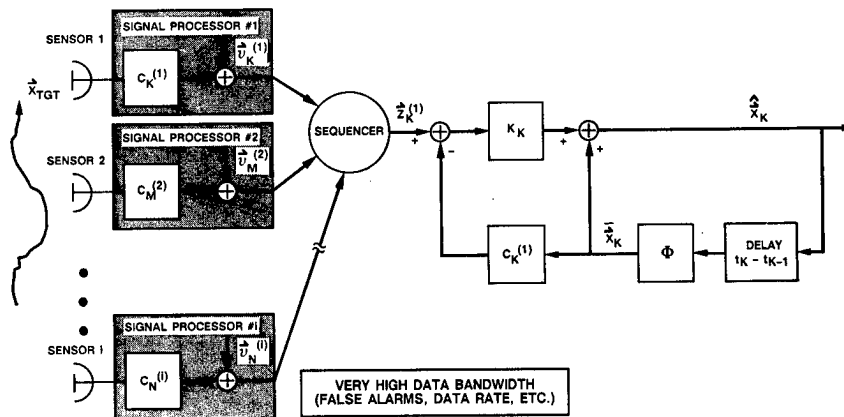
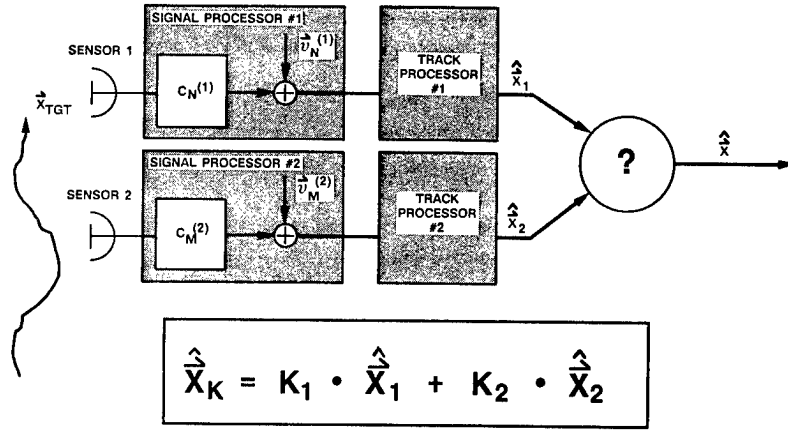


Figure 3. Measurement Fusion Solution



■ Objective: Linearly combine $\hat{\mathbf{x}}_1$ and $\hat{\mathbf{x}}_2$ to produce a MMSE, unbiased estimate of $\hat{\mathbf{x}}$

Figure 4. Track File Fusion Problem

where we've included the transt. nation C to account for the possibility that the first and second sensors' estimates may be in different spaces. Assuming the individual sensor estimates are unbiased (i.e., $E[\hat{\mathbf{e}}_1(k/n)] = E[\hat{\mathbf{e}}_2(k/m)] = 0$), we require

$$K_1 + K_2 C = I \quad (55)$$

for an unbiased composite estimate. Let

$$K_2 = K \quad (56)$$

then

$$K_1 = I - KC \quad (57)$$

and our composite state estimate error becomes

$$\hat{\mathbf{e}}(k) = (I - KC) \cdot \hat{\mathbf{e}}_1(k/n) + K \cdot \hat{\mathbf{e}}_2(k/m). \quad (58)$$

We want to solve for the matrix K that minimizes the mean length of $\hat{\mathbf{e}}(k)$ at every time instant. Once again, we want

$$\nabla_K E[\hat{\mathbf{e}}^T(k) \hat{\mathbf{e}}(k)] = [0]. \quad (59)$$

Writing the error as

$$\hat{\mathbf{e}}(k) = \hat{\mathbf{e}}_1(k/n) + K[\hat{\mathbf{e}}_2(k/m) - C\hat{\mathbf{e}}_1(k/n)] \quad (60)$$

and evaluating equation 59, we have

$$\nabla_K \left\{ E \left[\left(\hat{\mathbf{e}}_1^T(k/n) + [\hat{\mathbf{e}}_2^T(k/m) - \hat{\mathbf{e}}_1^T(k/n) C^T] K^T \right) \left(\hat{\mathbf{e}}_1(k/n) + K[\hat{\mathbf{e}}_2(k/m) - C\hat{\mathbf{e}}_1(k/n)] \right) \right] \right\} = [0] \quad (61)$$

or

$$\begin{aligned} \nabla_K \left\{ E \left[\hat{\mathbf{e}}_1^T(k/n) \hat{\mathbf{e}}_1(k/n) + \hat{\mathbf{e}}_1^T(k/n) K[\hat{\mathbf{e}}_2(k/m) - C\hat{\mathbf{e}}_1(k/n)] \right. \right. \\ \left. \left. + [\hat{\mathbf{e}}_2^T(k/m) - \hat{\mathbf{e}}_1^T(k/n) C^T] K^T \hat{\mathbf{e}}_1(k/n) \right. \right. \\ \left. \left. + [\hat{\mathbf{e}}_2^T(k/m) - \hat{\mathbf{e}}_1^T(k/n) C^T] K^T K[\hat{\mathbf{e}}_2(k/m) - C\hat{\mathbf{e}}_1(k/n)] \right] \right\} \\ = [0] \quad (62) \end{aligned}$$

Evaluating the derivatives first gives us (see equations 13-22)

$$\begin{aligned} 2 \left\{ E[\hat{\mathbf{e}}_1^T(k/n) \hat{\mathbf{e}}_2^T(k/m)] - E[\hat{\mathbf{e}}_1^T(k/n) \hat{\mathbf{e}}_1^T(k/n)] C^T \right\} \\ + 2 K \left\{ E[\hat{\mathbf{e}}_2^T(k/m) \hat{\mathbf{e}}_2^T(k/m)] - E[\hat{\mathbf{e}}_2^T(k/m) \hat{\mathbf{e}}_1^T(k/n)] C^T \right. \\ \left. - C E[\hat{\mathbf{e}}_1^T(k/n) \hat{\mathbf{e}}_2^T(k/m)] + C E[\hat{\mathbf{e}}_1^T(k/n) \hat{\mathbf{e}}_1^T(k/n)] C^T \right\} \\ = [0]. \quad (63) \end{aligned}$$

This is where measurement and track file fusion differ. In the measurement fusion case, the cross-terms went to zero because the error and

noise were zero-mean and uncorrelated. In the track file fusion case, the errors still have zero-mean, but they are most likely correlated. If the sensors initialize their estimates with the same initial value or if the same process model is assumed, then the sensor estimates will be correlated and the cross-terms are nonzero. Let

$$P_{12}(k) = E[\hat{\mathbf{e}}_1(k/n) \hat{\mathbf{e}}_2^T(k/m)], \quad (64)$$

$$P_{21}(k) = P_{12}^T(k) = E[\hat{\mathbf{e}}_2(k/m) \hat{\mathbf{e}}_1^T(k/n)], \quad (65)$$

$$P_{11}(k) = E[\hat{\mathbf{e}}_1(k/n) \hat{\mathbf{e}}_1^T(k/n)], \quad (66)$$

and

$$P_{22}(k) = E[\hat{\mathbf{e}}_2(k/m) \hat{\mathbf{e}}_2^T(k/m)], \quad (67)$$

then our requirement for a minimum error estimate is

$$\begin{aligned} P_{12}(k) - P_{11}(k) C^T + K[P_{22}(k) - P_{21}(k) C^T \\ - C P_{12}(k) + C P_{11}(k) C^T] = [0] \end{aligned} \quad (68)$$

and the optimal gain is

$$\begin{aligned} K = [P_{11}(k) C^T - P_{12}(k)] \cdot [P_{22}(k) \\ - P_{21}(k) C^T - C P_{12}(k) + C P_{11}(k) C^T]^{-1} \end{aligned} \quad (69)$$

Thus, if two sensors can supply filtered estimates of the target state vector, track file fusion is a two-step process. Update each sensor's estimate and error covariance to a common time and combine them with the composite gain as shown in Figure 5. From equation 49 and our expressions for K_1 and K_2 , the track file fusion algorithm is

$$\hat{\mathbf{x}}_K(k) = \hat{\mathbf{x}}_1(k/n) + K \cdot [\hat{\mathbf{x}}_2(k/m) - C \cdot \hat{\mathbf{x}}_1(k/n)] \quad (70)$$

$$\begin{aligned} K = [P_{11}(k) C^T + P_{12}(k)] [P_{22}(k) - P_{21}(k) C^T \\ - C P_{12}(k) + C P_{11}(k) C^T]^{-1} \end{aligned} \quad (71)$$

Unfortunately, accurate calculation of the gain K requires that we compute the cross-covariance term $P_{12}(k)$. If we initialize the cross-covariance term based on our initial state estimates, then we can update the term each time we perform a track file fusion update. Recall

$$P_{12}(k) = E[\hat{\mathbf{e}}_1(k/n) \hat{\mathbf{e}}_2^T(k/m)] \quad (72)$$

but from the measurement fusion derivation (equation 33), we know that

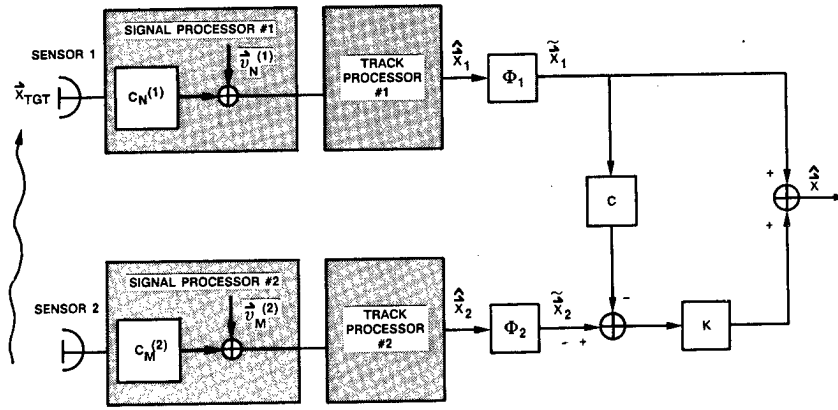


Figure 5. Track File Fusion Solution

$$\hat{\epsilon}_1(k/n) = \Phi_1(k-n) \hat{\epsilon}_1(n/n) - G_1 \hat{w}_1(n-1) \quad (73)$$

and

$$\hat{\epsilon}_2(k/m) = \Phi_2(k-m) \hat{\epsilon}_2(m/m) - G_2 \hat{w}_2(m-1). \quad (74)$$

Therefore, omitting the state transition arguments for brevity, we have

$$P_{12}(k) = E \left\{ [\Phi_1 \hat{\epsilon}_1(n/n) - G_1 \hat{w}_1(n-1)] [\hat{\epsilon}_2^T(m/m) \Phi_2^T - \hat{w}_2^T(m-1) G_2^T] \right\} \quad (75)$$

or

$$\begin{aligned} P_{12}(k) = & \Phi_1 \cdot E [\hat{\epsilon}_1(n/n) \hat{\epsilon}_2^T(m/m)] \cdot \Phi_2^T \\ & + G_1 E [\hat{w}_1(n-1) \hat{w}_2^T(m-1)] G_2^T \\ & - G_1 E [\hat{w}_1(n-1) \hat{\epsilon}_2^T(m/m)] \Phi_2^T \\ & - \Phi_1 E [\hat{\epsilon}_1(n/n) \hat{w}_2^T(m-1)] G_2^T. \end{aligned} \quad (76)$$

The cross-terms are zero, and if we let

$$P'_{12}(k-1) = E [\hat{\epsilon}_1(n/n) \hat{\epsilon}_2^T(m/m)] \quad (77)$$

and

$$Q'_{12}(k-1) = E [\hat{w}_1(n-1) \hat{w}_2^T(m-1)] \quad (78)$$

then the predicted cross-covariance, equation 76 becomes

$$P_{12}(k) = \Phi_1 P'_{12}(k-1) \Phi_2^T + G_1 Q'_{12}(k-1) G_2^T \quad (79)$$

The error covariance based on the predictions of $\hat{\epsilon}_1$ and $\hat{\epsilon}_2$ is (see measurement fusion equation 40)

$$\begin{aligned} P'_{12}(k-1) = & E \left\{ [(I-K_1 C_1) \hat{\epsilon}_1(n/n-1) + K_1 \hat{v}_1(n)] \right. \\ & \left. \cdot [\hat{\epsilon}_2^T(m/m-1) (I-K_2 C_2)^T + \hat{v}_2^T(m) K_2^T] \right\} \end{aligned} \quad (80)$$

where K_i , C_i , and \hat{v}_i are the i th sensor's gain, state-measurement transformation, and measurement noise vector, respectively. Since the measurement noises are uncorrelated

$$P'_{12}(k-1) = (I-K_1 C_1) E [\hat{\epsilon}_1(n/n-1) \hat{\epsilon}_2^T(m/m-1)] (I-K_2 C_2)^T \quad (81)$$

and if the time update intervals are synchronized, so that $n=m=k-1$, then from equation 72,

$$P'_{12}(k-1) = E [\hat{\epsilon}_1(n/n-1) \hat{\epsilon}_2^T(m/m-1)] \quad (82)$$

and we can recursively update the cross-error covariance matrix with equations 79 and 81

$$P_{12}(k) = \Phi_1 P'_{12}(k-1) \Phi_2^T + G_1 Q'_{12}(k-1) G_2^T \quad (83)$$

and

$$P'_{12}(k-1) = (I-K_1 C_1) P_{12}(k-1) (I-K_2 C_2)^T \quad (84)$$

or in one recursive equation (substitute equation 84 into equation 83)

$$\begin{aligned} P_{12}(k) = & \Phi_1 (I-K_1 C_1) P_{12}(k-1) (I-K_2 C_2)^T \Phi_2^T \\ & + G_1 Q_{12}(k-1) G_2^T \end{aligned} \quad (85)$$

With the above expression for $P_{12}(k)$ and the track file fusion equations 70 and 71, we can correctly fuse to track files. Unfortunately, besides the computational burden of computing $P_{12}(k)$, the algorithm requires that each sensor pass not only its error covariance matrix, $P_{11}(k)$ or $P_{22}(k)$, but also its state transition matrix, Kalman gain matrix, and process noise statistics. This additional data flow and the extra computational burden make the cross-covariance term extremely undesirable. If we could show that $P_{12}(k)$'s contribution is negligible, then the track file fusion algorithm (equations 70, 71, and 85) becomes

$$\hat{x}(k) = \hat{x}_1(k/n) + K [\hat{x}_2(k/m) - C \hat{x}_1(k/n)] \quad (86)$$

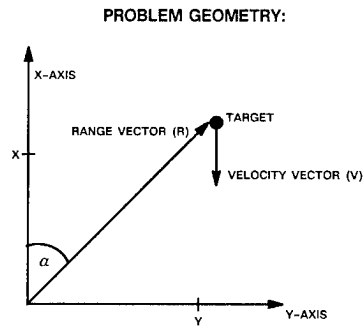
$$K = [P_{11}(k) C^T] \cdot [P_{22}(k) + C P_{11}(k) C^T]^{-1} \quad (87)$$

which is a much more manageable fusion algorithm.

2.3 Significance of $P_{12}(k)$ Term

To determine the significance of the $P_{12}(k)$ in track file fusion, we created a simulation that produced estimates with and without the cross-covariance term $P_{12}(k)$. Figure 6 summarizes our simulation. We simulated a system that included two independent sensors measuring independent Cartesian position components (that is, both sensors are located at the origin and they independently measure the position components x and y). Within a given sensor, the x and y measurements are independent of each other. We realize that this sensor suite is not representative of most applications, but we were mainly concerned with testing the significance of the term $P_{12}(k)$. We wanted as simple a simulation as possible that would still exercise the problem. We feel that the simulation exercises the $P_{12}(k)$ term better than most applications. Most applications have sensor suites consisting of dissimilar sensors (e.g., radar, IR, etc.), and their filtered outputs should be less correlated than a sensor suite containing very similar sensors. We initialized both sensors' estimates to the mean value of the initial state and gave them each the same process noise model; we believe that unless the filters use the same measurements, this situation is as correlated as any application. If the simulation shows that the contribution of the $P_{12}(k)$ is negligible, then we believe the term will be negligible in most applications.

The simulation generated one target at a random position with a random velocity. The initial position and velocity vectors were Rayleigh distributed in magnitude and uniformly distributed in direction. The simulation initialized the individual sensor filters (call them filter 1 and filter 2)



- Target appears randomly
 - Position - Velocity vectors are Rayleigh distributed in magnitude and uniformly distributed in direction
 - It continues at constant velocity/constant heading

SIMULATION:

- Two sensors
- Measure $[X \ Y]^T$ independently with different measurement noises
- Estimate $[X \ \dot{X} \ Y \ \dot{Y}]^T$ for both sensors
- Fuse estimates with and without correlation term
- Both filters initialized to statistical mean of target position and velocity vectors

Figure 6. Simulation Summary

to the mean values of the initial target state vector. It implemented a four-state position-velocity filter in each sensor. The only difference between filter 1 and filter 2 was that their input measurements were uncorrelated and had different statistics.

Once initialized, the target continued moving with a constant velocity vector (magnitude and direction) for the duration of the run. White process noise was added to the position state, and both sensor filters used the same process noise model. The two sensor estimates were fused with both track file fusion techniques (with and without the $P_{12}(k)$ term). The results from various runs are shown in Figures 7, 8, 9, and 10. The plots are root mean squared (RMS) error values from 100-point Monte Carlo runs. The results show that even when the process noise is high, the $P_{12}(k)$ contributes very little. On most of the scales it is impossible to distinguish between fusion with $P_{12}(k)$ and fusion without $P_{12}(k)$. Our conclusion is that the $P_{12}(k)$ contribution in track file fusion is not worth the extra processing required to compute the additional term. This is not to say that the term is insignificant in the association/correlation problem. Further investigation is required to determine whether exclusion of the cross-correlation term from the data association problem adversely affects tracking response. We are saying that once the association is complete, the effect of $P_{12}(k)$ on the optimal track file fusion gain is minimal, and if the term was omitted during the association process, then there is no reason to include it in the filtering update algorithm.

There is one drawback to the fusion technique discussed above. Look back at equations for track file fusion (#70 and #71). Note that the form is nonrecursive; it depends on the availability of data from both sensors at each iteration. If data is unavailable from sensor #2, then the gain approaches zero as $P_{22}(k)$ grows and the first sensor's estimate becomes the fused estimate. Alternatively, if data is unavailable from sensor #1, the gain approaches one and the second sensor's estimate becomes the fused estimate. Even though we may have been fusing data for a period of time before the inputs from one sensor become unavailable, that fused information is lost with the track file fusion form we have derived. Further study is required to determine whether a recursive track file fusion form should be employed or whether the current form is adequate.

In summary, if the measurements from each sensor are available and the data transfer bandwidth is acceptable, then the measurement fusion algorithm offers the best response. If, however, track files must be fused, then the track file fusion method without the cross-correlation term offers excellent performance. As a final note, we would like to establish that the curves presented in Figures 7, 8, 9, and 10 fail to illustrate the full benefits of fusion. Because we were looking for maximum correlation between sensors, the two sensors we simulated measure very similar parameters. The best fusion performance is achieved when each sensor in the multisensor system measures a different dimension of the target state (e.g., radar measures good range/range rate, and infrared measures good angle).

3.0 REFERENCES

- [1] Bar-Shalom, Y. "On the Track-to-Track Correlation Problem," *IEEE Transactions of Automatic Control*, April 1981, pp. 571-572.
- [2] Gelb, Arthur ed. *Applied Optimal Estimation*, The MIT Press, Cambridge, Mass., 1974.

- [3] Van Trees, Harry L. *Detection, Estimation, and Modulation Theory - Part I*. Wiley & Sons, New York, 1968.
- [4] Yannoni, Ronald. "UD Factorization Applied to Airborne Kalman-Filter-Based Fusion." *NAECON '88*, Dayton, Ohio, 1988.

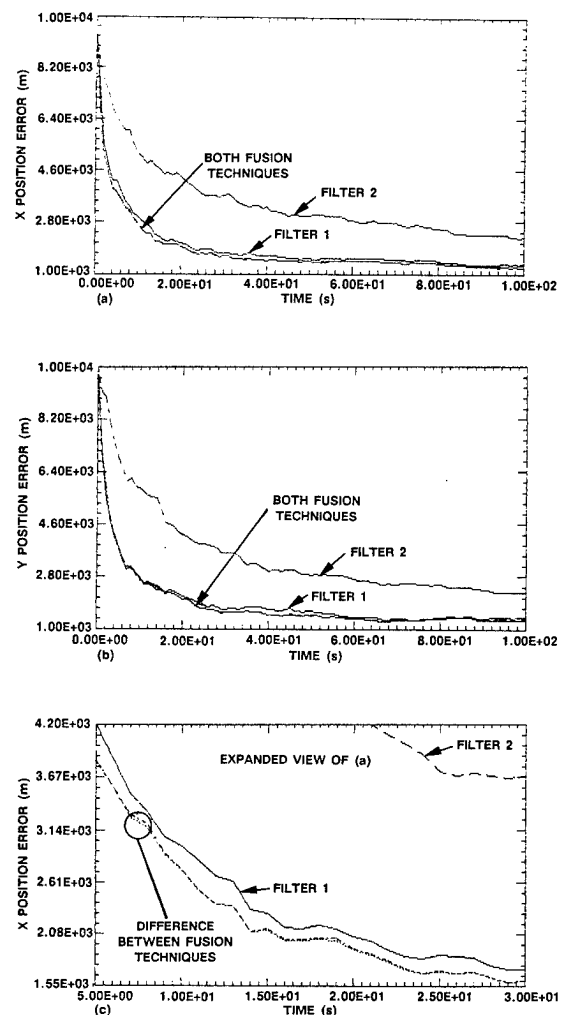


Figure 7. High Measurement Noise, Low Process Noise

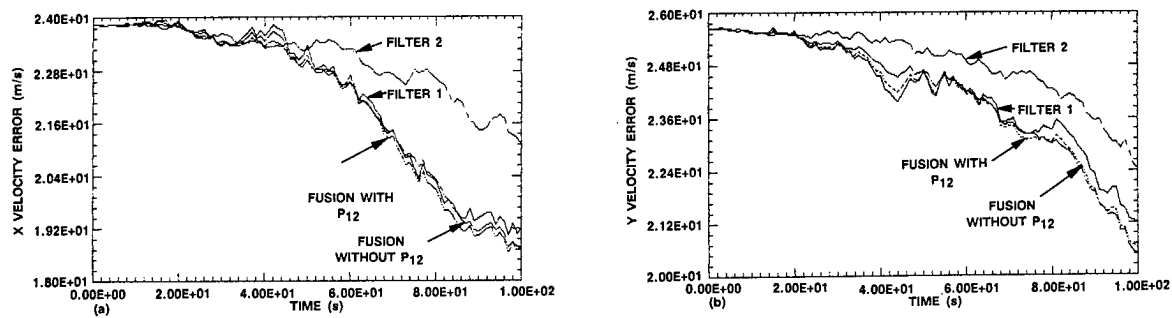


Figure 9. High Measurement Noise, Medium Process Noise

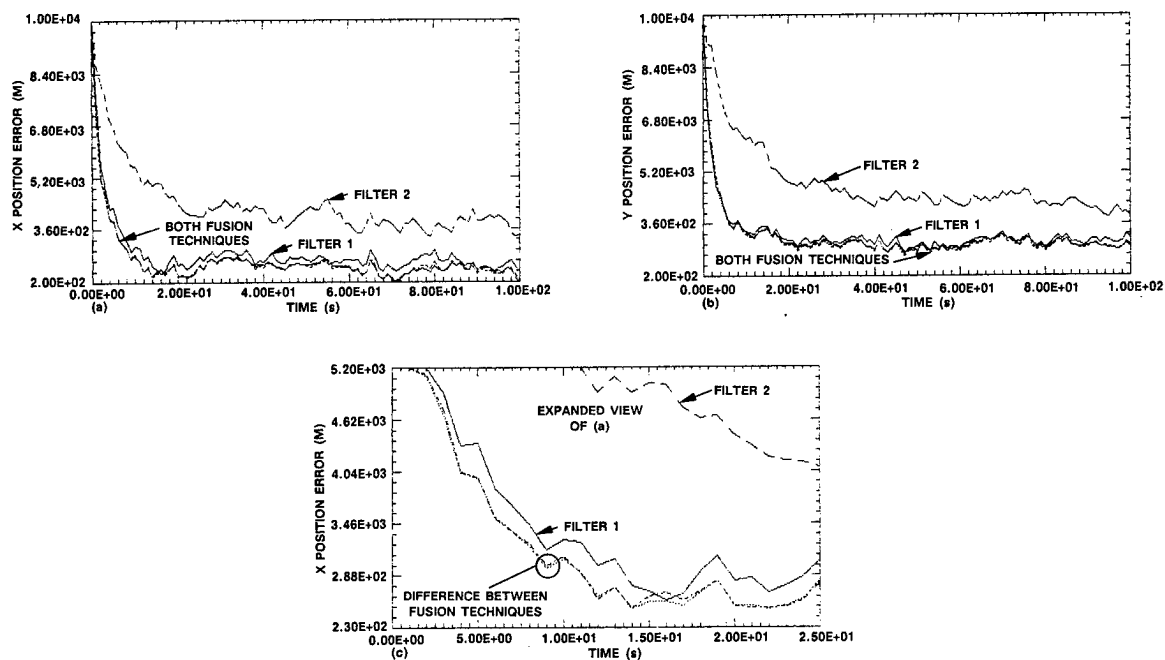


Figure 8. High Measurement Noise, Low Process Noise

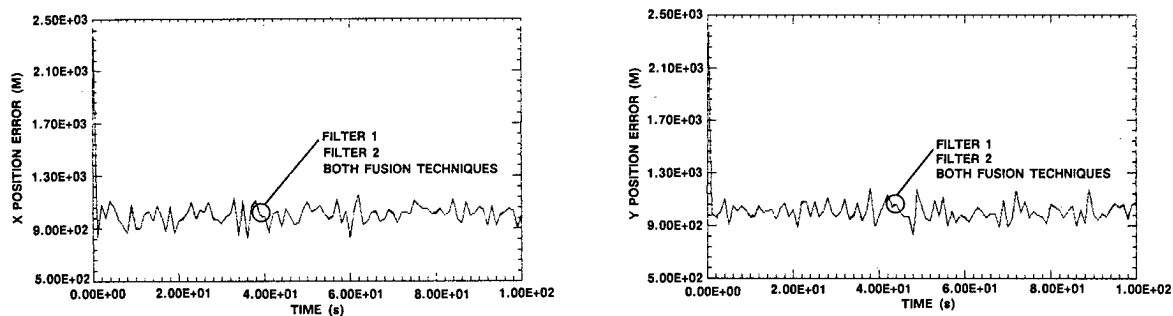


Figure 10. High Process Noise, Low Measurement Noise

THIS PAGE LEFT BLANK INTENTIONALLY

TRACK INITIALIZATION AS AN INTEGER PROGRAM WITH POSSIBLE TARGET DETECTIONS

Scott L. Godfrey

FMC Corporation
East Coast Engineering Office
1 Danube Drive
King George, VA 22485

The paper describes an algorithm for solving multitarget tracking problems in the presence of false alarms and missed target detections. The algorithm is an extension of an algorithm found in the literature which did not consider tracking environments where the probability of detection was less than one. The algorithm uses a Kalman filter to generate a set of candidate target tracks from received measurements. Infinitely erroneous measurements called auxiliary measurements are used to accomodate false alarms and missed detections. A method is described to estimate a bound on the number of targets present (n_T) in the surveillance area from the number of measurements received and *a priori* models for the number of targets present, the number of false alarms, and the probability of detection. Integer 0-1 programming is used to pick the n_T best tracks from the set of candidate tracks. Some simulation results are given to demonstrate the operation of the algorithm.

TRACK INITIALIZATION AS AN INTEGER PROGRAM WITH POSSIBLE TARGET DETECTIONS

Scott L. Godfrey

FMC Corporation
East Coast Engineering Office
1 Danube Drive
King George, VA 22485

I. INTRODUCTION

Simply stated, multitarget tracking is the determination of target trajectory estimates or tracks from measurement data originating from the targets. The trajectory estimates consist of parameters such as position, velocity, identity, etc., and the measurements can consist of target positions and velocities as well as other quantities. Once the measurements have been received, the problem is not so much in updating the individual tracks with the received measurements as it is in deciding with which tracks the measurements are associated. Two complicating factors are that some of the received measurements can be false alarms, and the probability of detecting a target can be less than one. This measurement-to-track correlation problem is the primary difficulty in multitarget tracking.

Much has been written about the multitarget tracking problem, and most of it can be found in the survey texts by Bar-Shalom [ref. 2] and Blackman [ref. 3]. The methods used in addressing the measurement-to-track correlation problem have generally involved the use of Bayesian, maximum likelihood, and maximum *a posteriori* estimation techniques. Using the Bayesian approach to track estimation, a target track is updated using a weighted sum of measurements which are close enough in a statistical sense to the target. The weight assigned to a measurement is the probability (likelihood) that the correlation hypothesis, "the measurement corresponds to the track," is correct. Bar-Shalom's Joint Probabilistic Data Association Filter (JPDAF) [ref. 1] and Reid's multiple-hypothesis filter [ref. 6] are examples of algorithms which use the Bayesian approach.

In contrast, algorithms using the maximum likelihood (ML) estimation method or the maximum *a posteriori* (MAP) estimation method update a track using the single measurement which is statistically closest to the target, or equivalently, they choose the correlation hypothesis which is most likely. The difference between the two techniques, ML and MAP, is that the MAP technique assumes prior knowledge about the likelihood of the different hypotheses but the ML method does not. The ML and MAP estimation methods are similar in that each makes a "hard" decision as to which measurement is correct. On the other hand, the Bayesian estimation makes a "soft" decision where measurements with large likelihoods dominate the multitarget tracking solution, but measurements with small likelihoods still have some influence on the solution.

Morefield's integer programming algorithm [ref. 5] is an example of the ML/MAP estimation methods. Morefield casts the multitarget tracking problem in the framework of pattern recognition. He uses a Kalman filter to generate a set of candidate tracks from received measurements. Each of the candidate tracks has an acceptable goodness-of-fit to an assumed dynamic model and is considered not too unlikely to be of an actual target. Integer programming is used to select the best set of tracks from the set of candidate tracks so that each received measurement is used in the computation of only one track.

Morefield allowed for the possibility that received measurements may be false alarms. However, he did not consider tracking environments where the probability of detecting a target is less than one. This paper presents an extension to Morefield's tracking method which allows for a nonunity probability of detection. The algorithm presented is a batch algo-

rithm, and due to the rapidly growing computational burden associated with it as the number of sensor scans increases, its main application is to track initiation where the number of sensor scans is relatively small.

The paper is organized as follows. In section II, the dynamic target model and the Kalman filter equations are presented briefly. Morefield's tracking method is presented in section III. The auxiliary measurement idea is presented in section IV with a discussion of how Morefield's algorithm can be modified to use auxiliary measurements to accommodate both missed detections and false alarms. In section V, a method for estimating a bound (n_T) on the number of targets in the surveillance area is presented. The algorithm picks the n_T best tracks from the set of candidate tracks as shown in the simulation results presented in section VI. Conclusions are given in section VII.

II. THE DYNAMIC TARGET MODEL AND THE KALMAN FILTER

For the development, it is assumed that target behavior is described by the linear stochastic system

$$x_{k+1} = \Phi_{k+1,k}x_k + w_k, \quad (1)$$

$$z_k = Hx_k + v_k, \quad (2)$$

where x_k is the $n \times 1$ state vector at time t_k and contains the target parameters to be estimated; $\Phi_{k+1,k}$ is the transition matrix which describes how the state of the system evolves with time, and w_k is zero-mean Gaussian process noise with covariance Q_k . The $m \times 1$ measurement z_k is of the state x_k , and H is the measurement matrix which describes how the state and measurement vectors are related. The error in the measurement is v_k which is zero-mean, Gaussian, and has covariance R_k . Also it is assumed that the initial state of the system x_0 is a Gaussian random vector with mean \bar{x}_0 and covariance P_0 .

If the correspondence between received measurements and tracked targets is known, then trajectory estimates can be generated using the well-known Kalman filter equations:

Prediction:

$$\hat{x}_{k+1/k} = \Phi_{k+1,k}\hat{x}_{k/k}, \quad (3)$$

$$\Sigma_{k+1/k} = \Phi_{k+1,k}\Sigma_{k/k}\Phi_{k+1,k}^T + Q_k. \quad (4)$$

Correction:

$$\hat{x}_{k/k} = \hat{x}_{k/k-1} + K_k \tilde{z}_k, \quad (5)$$

$$\tilde{z}_k = z_k - H\hat{x}_{k/k-1}, \quad (6)$$

$$\Sigma_{k/k} = (I - K_k H)\Sigma_{k/k-1}. \quad (7)$$

Kalman Gain:

$$K_k = \Sigma_{k/k-1}H^T V_k^{-1}, \quad (8)$$

$$V_k = H\Sigma_{k/k-1}H^T + R_k. \quad (9)$$

Initialization:

$$\hat{x}_{0/-1} = \bar{x}_0, \quad (10)$$

$$\Sigma_{0/-1} = P_0. \quad (11)$$

The subscript m/n denotes a filter quantity at time t_m given measurements received up to time t_n . Thus $\hat{x}_{k/k}$ denotes the estimate at time t_k given measurements received up to time t_k . The quantity $\Sigma_{k/k}$ is the covariance of the estimation error associated with $\hat{x}_{k/k}$. The measurement residual or innovation \tilde{z}_k is the difference between the measurement z_k and the predicted measurement $H\hat{x}_{k/k-1}$. It has a covariance V_k .

III. THE FORMATION OF TARGET TRACKS: MOREFIELD'S ALGORITHM

Morefield uses a multiple hypothesis approach to form tracks from received measurements. Each hypothesis is the event that a particular set of measurements originated from a single target and can therefore be used to compute a target track. Let $z(i)$ represent the measurements received on scan i and let n_i be the number of measurements received on scan i . It is assumed that all data received on the same scan are received at the same time. Also, let $Z^k = \{z(1), \dots, z(k)\}$ represent the collection of data received through scan k , and $Z^{k,l} = \{z_{\nu_1}, \dots, z_{\nu_k}\}$ be a set of measurements, one chosen from each sensor scan. The subscript ν_i serves to indicate which of the measurements received on scan i is included in the set.

From the set of measurements $Z^{k,l}$, a target track can be generated using the Kalman filter. A chi-square gating criterion must be satisfied for each measurement in $Z^{k,l}$ before $Z^{k,l}$ can be considered to form a candidate target track:

$$\tilde{z}_{\nu_i}^T V_{\nu_i}^{-1} \tilde{z}_{\nu_i} < \gamma, \quad (12)$$

where γ is such that

$$P(\tilde{z}_{\nu_i}^T V_{\nu_i}^{-1} \tilde{z}_{\nu_i} > \gamma) = \alpha, \quad (13)$$

and α is small. Inequality (12) describes an m -dimensional hyperellipsoid, the interior of which defines the surveillance region at time t_i in the measurement space. In addition to the requirement in (12), Morefield requires that the entire track formed satisfy a quality constraint so that

$$P(Z^{k,l} \text{ forms a false track}) < \alpha' \quad (14)$$

for a given α' . Let Ω be the set of sequences of measurements which are not too unlikely to form tracks of actual targets.

Denote $\tau = \{Z^{k,l_j}\}_{j=0}^{n_T}$ as a feasible partition of the measurements where n_T is the number of targets present in the surveillance area. A feasible partition has the following properties:

$$\tau \subset \Omega; \quad (15)$$

$$Z^k = \bigcup_{j=0}^{n_T} Z^{k,l_j}, \quad (16)$$

and

$$Z^{k,l_i} \cap Z^{k,l_j} = \emptyset, \quad \text{if } l_i \neq l_j. \quad (17)$$

The sequence Z^{k,l_0} contains measurements which cannot be associated with a track for the given partition.

The tracks considered best are those associated with the most likely partition which in this case is the one with the maximum posterior probability $p(\tau|Z^k)$. Using Bayes' rule

$$p(\tau|Z^k) = \frac{p(Z^k|\tau)p(\tau)}{p(Z^k)}, \quad (18)$$

where, $p(\tau)$ is the *a priori* probability that a particular partition is correct. Ignoring the normalizing factor $p(Z^k)$, the tracks considered best are associated with the partition which maximizes $p(Z^k|\tau)p(\tau)$.

As stated, the optimal partition is the maximum *a posteriori* estimate. However, if the prior probability density function (pdf) $p(\tau)$ is uniform over all possible partitions, i.e. there is no prior information, then the optimal partition maximizes the likelihood function $p(Z^k|\tau)$ and provides a maximum likelihood estimate of the target tracks.

The problem of maximizing (18) can be transformed into a more convenient minimization problem by using the negative log of $p(Z^k|\tau)p(\tau)$, the cost of the partition, as a measure of the likelihood of the partition. Assuming tracks are independent

$$p(Z^k|\tau) = \prod_{j=0}^{n_T} p(Z^{k,l_j}|\tau), \quad (19)$$

and the negative log-likelihood function is the cost

$$J = -\ln[p(\tau)] - \sum_{j=0}^{n_T} \ln[p(Z^{k,l_j}|\tau)]. \quad (20)$$

As shown in [ref. 5],

$$-\sum_{j=0}^{n_T} \ln[p(Z^{k,l_j}|\tau)] = \sum_{j=1}^{n_T} \psi^{k,l_j} \quad (21)$$

where

$$\psi^{k,l_j} = \sum_{i=1}^k \frac{1}{2} \left[\ln|V_{\nu_i}| + \tilde{z}_{\nu_i}^T V_{\nu_i}^{-1} \tilde{z}_{\nu_i} + m \ln 2\pi \right] - \ln Y_{\nu_i}, \quad (22)$$

and Y_{ν_i} is the volume of the surveillance region defined by inequality (12),

$$Y_{\nu_i} = \frac{(\gamma\pi)^{\frac{m}{2}} |V_{\nu_i}|^{\frac{1}{2}}}{\Gamma\left(\frac{m}{2} + 1\right)}, \quad (23)$$

and $\Gamma(\cdot)$ is the gamma function. Substituting (23) into (22) simplifies ψ^{k,l_j} to

$$\psi^{k,l_j} = \sum_{i=1}^k \frac{1}{2} \tilde{z}_{\nu_i}^T V_{\nu_i}^{-1} \tilde{z}_{\nu_i} - k \left[\frac{m}{2} \ln \frac{\gamma}{2} - \ln \Gamma\left(\frac{m}{2} + 1\right) \right]. \quad (24)$$

The selection of an expression for $p(\tau)$ is somewhat subjective. Morefield suggests several possibilities. However, since our ultimate goal is to initiate target tracks in the presence of false alarms and missed target detections, we would like $p(\tau)$ to be such that a track formed with a large number of measurements has a larger likelihood than a track formed with a small number of measurements. For this application, the form of the prior probability is

$$p(\tau) = \prod_{i=1}^{n_T} \frac{1}{c_n} e^{n_{l_j}^2}, \quad (25)$$

where n_{l_j} is the number of measurements in Z^{k,l_j} and c_n is the normalization constant

$$c_n = \sum_{i=0}^k e^{i^2}. \quad (26)$$

If the cost contribution in (21) is ignored, the form of $p(\tau)$ makes it more desirable to have in a partition one track with k measurements rather than two tracks with k_1 and k_2 measurements where $k_1 + k_2 = k$. Combining the expression in (25) with the expression in (24), yields the cost c^{k,l_j} associated with Z^{k,l_j}

$$c^{k,l_j} = \ln c_n - n_{l_j}^2 + \sum_{i=1}^{n_{l_j}} \frac{1}{2} \tilde{z}_{\nu_i}^T V_{\nu_i}^{-1} \tilde{z}_{\nu_i} - n_{l_j} \left[\frac{m}{2} \ln \frac{\gamma}{2} - \ln \Gamma\left(\frac{m}{2} + 1\right) \right]. \quad (27)$$

Morefield's solution to the problem is as follows. Assume that there are L elements in the set Ω of feasible measurement sequences. Let λ^{k,l_j} be a binary measurement indicator vector of dimension N_{meas} , the total number of measurements received. Ones indicate which measurements belong to Z^{k,l_j} , and all other elements are zeros. Then, an $N_{\text{meas}} \times L$ binary matrix A can be used to represent Ω . Each column of A is a measurement indicator vector corresponding to one of the sequences in Ω . The costs associated with each column of A can be represented in a $1 \times L$ vector c_v . Using an $L \times 1$ binary partition indicator vector p to indicate which columns of A are included in a

particular partition τ , the problem of finding the optimal partition can be equivalently expressed as the 0-1 integer programming problem:

$$\min_{\rho} c_v \cdot \rho \quad (28)$$

subject to the inequality constraint

$$A \cdot \rho \leq 1, \quad (29)$$

where 1 is a vector of ones. Equation (28) and inequality (29) describe a set packing problem [ref. 4]. Note that the measurements not used in the formation of tracks as indicated by the optimal partition are the false alarms. If there are no false alarms, then the constraint in (29) becomes

$$A \cdot \rho = 1 \quad (30)$$

requiring each measurement to be assigned to a track, a set partitioning problem [ref. 4].

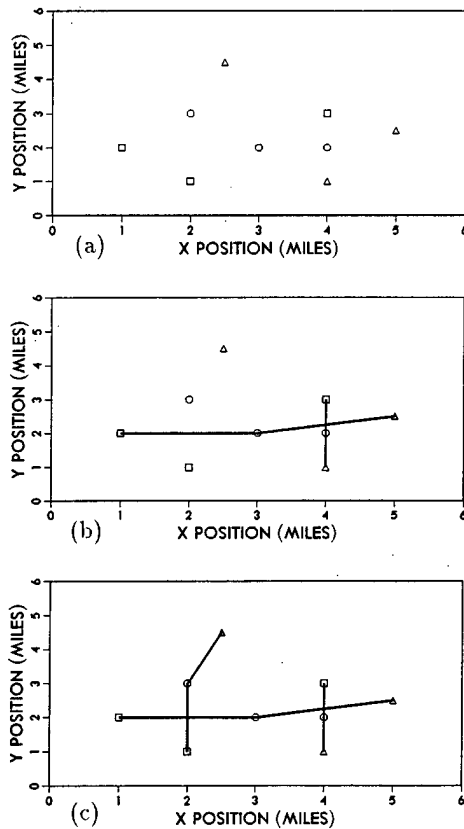


Figure 1. Three track formation hypotheses. (a) No targets present. (b) Two targets present. (c) Three targets present.

An illustration of how tracks might be formed from measurements is shown in Figure 1. Like symbols are used to represent measurements received on the same sensor scan. Shown in Figure 1(a) are measurements received from three sensor scans. Figure

1(a) also represents the hypothesis that there are no targets present; all measurements are false alarms ($A \cdot \rho = 0$). Figure 1(b) depicts the hypothesis that two targets are present, and there are three false alarms (three zeros in $A \cdot \rho$). Figure 1(c) depicts the hypothesis that three targets are present and there are zero false alarms ($A \cdot \rho = 1$).

IV. THE FORMATION OF TARGET TRACKS: AUXILIARY MEASUREMENTS

If the probability of detection is less than one, then the number of hypotheses which must be considered in forming tracks is infinite. For example, in Figure 1, a large number of targets could be present in the surveillance region, but the sensors fail to detect them. Although, in most situations, this is unlikely. The approach presented in this section is to account for the missed detections by completing the surveillance picture with artificial measurements called *auxiliary measurements*. When a target fails to be detected, an auxiliary measurement is used to update the target's track. The auxiliary measurements are infinitely erroneous and therefore have no influence on tracks formed with them. Using this approach, the multitarget tracking problem can be solved as a set partitioning problem using efficient integer programming techniques.

In the operation of a track processing system, the Kalman filters used for state estimation extrapolate corrected target estimates to predict where targets are likely to be when measurements are taken, and the predicted estimates are updated with the received measurements. If no measurement is received from a target (a missed detection), the track processor skips the correction phase of the Kalman filter and predicts where the target is likely to be the next time measurements are taken. Eventually, either a new measurement is received and the target's track is updated, or the track becomes stale, and it is terminated.

Skipping the correction phase of the Kalman filter is equivalent to setting the Kalman gain equal to zero and updating the predicted estimate as if a measurement had been received. Define an auxiliary measurement as a measurement which contains no information, i.e., the covariance of the measurement error associated with it is

$$R = \text{diag} [\infty, \dots, \infty]. \quad (31)$$

This makes the covariance of the measurement residual

$$V_k = H \Sigma_{k/k-1} H^T + R_k \quad (32)$$

infinite and the Kalman gain zero. The auxiliary measurement is infinitely erroneous so that updating a target track with it is equivalent to declaring a missed detection of the target. Naturally, since an auxiliary measurement acts as a missed detection, its

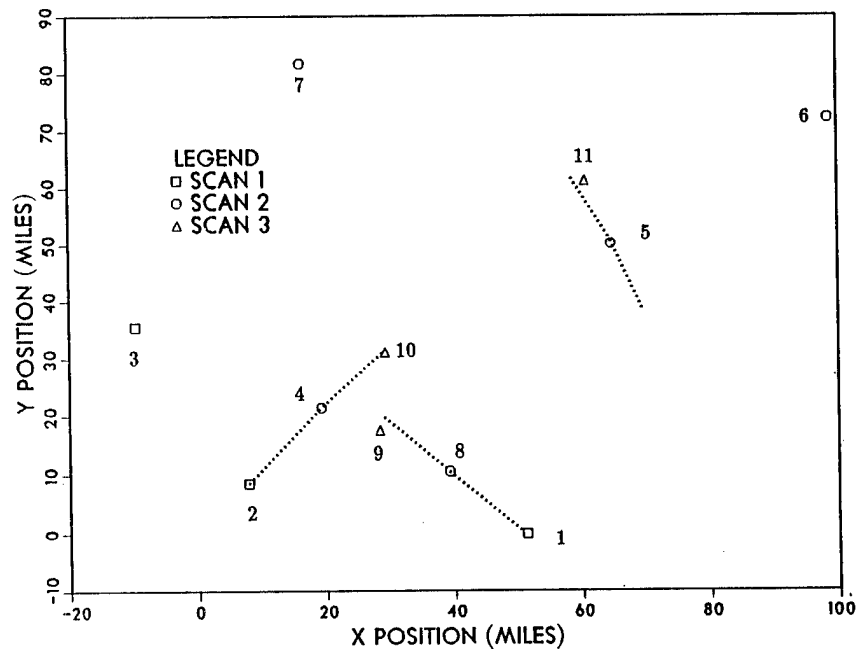


Figure 2. Track formation for $P_d < 1$.

presence is not included in the cost computation of equation (27).

The auxiliary measurement idea is a device which fits nicely into the appealing prediction/correction structure of the Kalman filter. With it, there is always a measurement with which to update a track. In the present approach to track initiation, auxiliary measurements are used to accommodate both missed detections and false alarms. The underlying philosophies are that first, an auxiliary measurement will be used to fill in a track when a missed detection occurs, and second, false alarms will be isolated in tracks consisting predominately of auxiliary measurements. Heuristically, the first point follows because an auxiliary measurement has the same effect on a track as a missed detection. The second point follows because to the track initiation procedure, a false alarm is a detection of a target which reports on only a single scan. Auxiliary measurements are used to account for the missed detections of the "target" on the other sensor scans.

To see how the track initiation problem can be transformed to a set partitioning problem, assume that the supply of auxiliary measurements is infinite and that the constraint in inequality (14) is relaxed. In this case, every received measurement can be placed in a track computed from itself and a collection of auxiliary measurements. The tracks are feasible because the erroneous nature of an auxiliary measurement ensures that inequality (12) is satisfied. Although these tracks are trivial, their existence en-

sures that there is at least one partition of the measurements which assigns every measurement to exactly one track.

Table 1.
A Measurement Partition for Data in Figure 2.

TRACK NUMBER	MEASUREMENT NUMBER	MEASUREMENT SCAN: (x, y)
1	2	1 : (7.8088, 8.4710)
	4	2 : (19.2519, 21.4220)
	10	3 : (29.3275, 31.0576)
2	1	1 : (51.1203, -0.1941)
	8	2 : (39.2674, 10.5393)
	9	3 : (28.4304, 17.49249)
3	5	1 : AM
	11	2 : (64.6668, 50.1273)
		3 : (60.6907, 61.1080)
4	3	1 : (-9.6299, 35.4111)
	7	2 : (16.3081, 81.7034)
		3 : AM
5	6	1 : AM
		2 : (98.6156, 72.1212)
		3 : AM

As an example of how feasible tracks can be formed using auxiliary measurements, consider the situation in Figure 2. The figure shows measurements from three targets received over three sensor scans with a probability of detection of 0.75. The actual target tracks are indicated by the dotted lines. False alarms are assumed to be uniformly distributed in

the surveillance area where the number occurring on a scan is modeled as Poisson

$$P(n \text{ FA}) = \frac{(\lambda Y)^n e^{-\lambda Y}}{n!}, \quad (33)$$

where Y is the volume of the surveillance area and λ is the mean number of false alarms per unit volume of surveillance area. Here, λY was chosen as 1. A partition of the measurements showing the measurements used to generate five candidate target tracks is shown in Table 1. The measurement numbers correspond to those in Figure 2. Of course, this is not the only measurement partition.

V. ESTIMATING THE MAXIMUM NUMBER OF TARGETS PRESENT

In this section, the aim is to compute a bound on the number of targets in the surveillance area so that a complete partitioning of the measurements is not required. The bound is computed based upon the number of measurements received on each sensor scan and *a priori* models for the number of targets present in the surveillance area, the number of false alarms, and the probability of detection. The goal is to speed the solution of the problem by dropping the requirement that the measurements be partitioned, and instead select the n_T best target tracks as a solution.

Assume measurement data are received for N sensor scans. Assume that the number of targets present during the reception of measurements is constant, i.e. targets neither disappear nor appear during the N sensor scans. It is required that n_T be computed such that

$$P(\text{number of targets} > n_T | N \text{ scans of data}) < \alpha \quad (34)$$

where α is small. Let n_i represent the number of measurements (target detections and false alarms) received on scan i , and $\{n_j\}_{j=1}^N$ be the set indicating the number of measurements received on the N sensor scans. Then an equivalent problem is to compute n_T such that

$$P(\text{number of targets} > n_T | \{n_j\}_{j=1}^N) < \alpha. \quad (35)$$

Let N_T represent the random variable for the number of targets present. Then

$$P(N_T > n_T | \{n_j\}_{j=1}^N) = 1 - P(N_T \leq n_T | \{n_j\}_{j=1}^N) \quad (36)$$

and

$$P(N_T \leq n_T | \{n_j\}_{j=1}^N) = \sum_{k=0}^{n_T} P(k | \{n_j\}_{j=1}^N) \quad (37)$$

where $P(k | \{n_j\}_{j=1}^N)$ is the probability that k targets are present given the received data. Using Bayes' rule

and the total probability theorem

$$P(k | \{n_j\}_{j=1}^N) = \frac{P(\{n_j\}_{j=1}^N | k) P(k)}{\sum_{m=0}^{\infty} P(\{n_j\}_{j=1}^N | m) P(m)} \quad (38)$$

where $P(k)$ is the *a priori* probability that k targets are present in the surveillance region. A Poisson model is used for $P(k)$

$$P(k) = \frac{(\kappa Y)^k e^{-\kappa Y}}{k!}. \quad (39)$$

Here, Y is the volume of the surveillance area and κ is the mean number of targets per unit volume of surveillance. The mean number of targets present is κY .

The expression for $P(\{n_j\}_{j=1}^N | k)$ is needed next. Since the number of targets present is constant, the detection experiments are independent on each scan. Therefore,

$$P(\{n_j\}_{j=1}^N | k) = \prod_{j=1}^N P(n_j | k). \quad (40)$$

If $n_j \leq k$,

$$P(n_j | k) = \sum_{i=0}^{n_j} P(i \text{ target detections}, n_j - i \text{ FA} | k). \quad (41)$$

If $n_j > k$,

$$P(n_j | k) = \sum_{i=0}^k P(i \text{ target detections}, n_j - i \text{ FA} | k). \quad (42)$$

Equation (42) follows because there can be no more than k target detections if there are k targets. Assuming that false alarms are independent of the number of targets present

$$P(i \text{ target detections}, n_j - i \text{ FA} | k) = P(i \text{ target detections} | k) P(n_j - i \text{ FA}). \quad (43)$$

The number of false alarms received is assumed to obey a Poisson model

$$P(n_j - i \text{ FA}) = \frac{(\lambda Y)^{n_j - i} e^{-\lambda Y}}{(n_j - i)!}, \quad (44)$$

and the probability of i target detections is given by the binomial probability

$$P(i | k) = \binom{k}{i} p_d^i (1 - p_d)^{k-i}. \quad (45)$$

Using (43) and substituting (44) and (45) into (41) and (42) yields for $n_j \leq k$

$$P(n_j | k) = \sum_{i=0}^{n_j} \binom{k}{i} p_d^i (1 - p_d)^{k-i} \frac{(\lambda Y)^{n_j - i} e^{-\lambda Y}}{(n_j - i)!}, \quad (46)$$

and for $n_j > k$

$$P(n_j|k) = \sum_{i=0}^k \binom{k}{i} p_d^i (1-p_d)^{k-i} \frac{(\lambda Y)^{n_j-i} e^{-\lambda Y}}{(n_j-i)!}. \quad (47)$$

Substituting (46) and (47) into (40) yields

$$P(\{n_j\}_{j=1}^N | k) = \prod_{j=1}^N \sum_{i=0}^{\min(k, n_j)} \binom{k}{i} p_d^i (1-p_d)^{k-i} \frac{(\lambda Y)^{n_j-i} e^{-\lambda Y}}{(n_j-i)!}. \quad (48)$$

Equation (48) can be used in equations (37) and (38) to compute n_T for a given α . Obviously, for any implementation the sum in the denominator of (38) must be truncated.

Table 2.
Bound on the Number of Targets
for Several Conditions

	(λY)	(κY)	(P_d)	CONFIDENCE	N_T
1	1.0	4.0	0.9	90 %	4
2	1.0	4.0	0.7	90 %	5
3	3.0	4.0	0.9	90 %	3
4	1.0	2.0	0.9	90 %	4
5	1.0	6.0	0.9	90 %	4
6	5.0	4.0	0.9	90 %	2
7	1.0	4.0	0.5	90 %	7
8	1.0	4.0	0.9	99 %	5

Examples for n_T are shown in Table 2. Data was received for five sensor scans where $\{n_i\}_{i=1}^5 = \{4, 5, 3, 5, 4\}$. The sum in the denominator of equation (38) was truncated at $m = 20$. For the case shown, the top row in the table represents n_T for a nominal set of parameters (mean no. false alarms λY , mean no. targets κY , probability of detection P_d , and confidence level). For this case, as seen from rows 2, 3, 6, and 7, n_T is most sensitive to the assumed false alarm rate, and the probability of detection.

VI. SIMULATION RESULTS

To limit the number of candidate target tracks, a simple m -contacts-out-of- N -sensor-scans test was used to accomplish the track quality test in inequality (14) where the number m depended upon the probability of detection. Five constant velocity targets were observed over five sensor scans under varying conditions. The targets were tracked in a Cartesian (x, y) coordinate frame, and were assumed to travel at constant speed with process noise accounting for deviations from this behavior. The motion model was

$$\begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix}_{k+1} = \begin{bmatrix} 1 & 0 & T & 0 \\ 0 & 1 & 0 & T \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix}_k + \begin{bmatrix} w_x \\ w_y \\ w_x \\ w_y \end{bmatrix}_k. \quad (49)$$

The dimensions were miles for positions and miles per hour for speeds. A diffuse model was used for the initial state ($P_0 = \text{diag}[\infty \ \infty \ \infty \ \infty]$), and the covariance of the process noise was

$$Q_k = \begin{bmatrix} \frac{T^3 \sigma^2}{3} & 0 & \frac{T^2 \sigma^2}{2} & 0 \\ 0 & \frac{T^3 \sigma^2}{3} & 0 & \frac{T^2 \sigma^2}{2} \\ \frac{T^2 \sigma^2}{2} & 0 & T \sigma^2 & 0 \\ 0 & \frac{T^2 \sigma^2}{2} & 0 & T \sigma^2 \end{bmatrix}, \quad (50)$$

where T , the time between sensor scans, was chosen as 1 hour and σ^2 was chosen as 0.25. The form of the covariance in equation (50) is derived by assuming that the process noise is the system response to a continuous time white noise input. Synthetic measurements of target positions were generated according to

$$\begin{bmatrix} z_x \\ z_y \end{bmatrix}_k = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ \dot{x} \\ \dot{y} \end{bmatrix}_k + \begin{bmatrix} v_x \\ v_y \end{bmatrix}_k, \quad (51)$$

where the measurement error was Gaussian with covariance

$$R = \text{diag}[1.0 \ 1.0]. \quad (52)$$

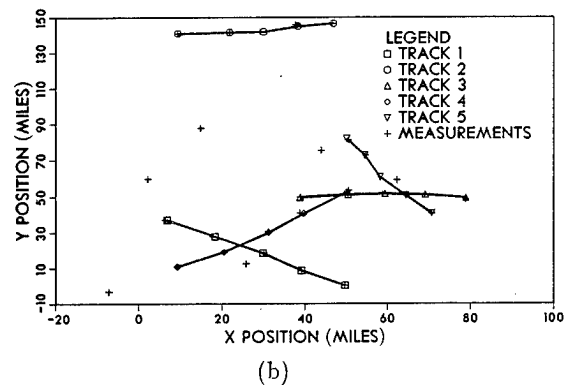
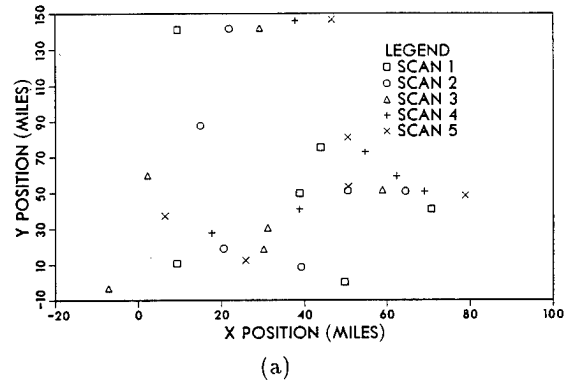


Figure 3. Track Formation for Scenario 1. (a) Received measurements. (b) Initialized tracks.

In the first scenario, the probability of detection was chosen as 0.95, the false alarm rate (λY) as 1 per

scan, and the *a priori* mean number of targets present (κY) as 3. The data received are shown in Figure 3(a), and the bound (n_T) on the number of targets present was computed as 6 with a confidence of 90%. To be a candidate track, 4 contacts had to be received over the 5 sensor scans in addition to having to satisfy inequality (12) for $\gamma = 9.21$, the 99% threshold for a two degree-of-freedom chi-square random variable. The tracks initiated by the procedure are shown in Figure 3(b). The tracks are correct.

In the solution to the problem only 5 tracks were initiated, but n_T was computed to be 6. The missing track consists entirely of auxiliary measurements. Tracks consisting of auxiliary measurements are counted as candidate tracks in order to "take up the slack" in the integer programming problem if needed.

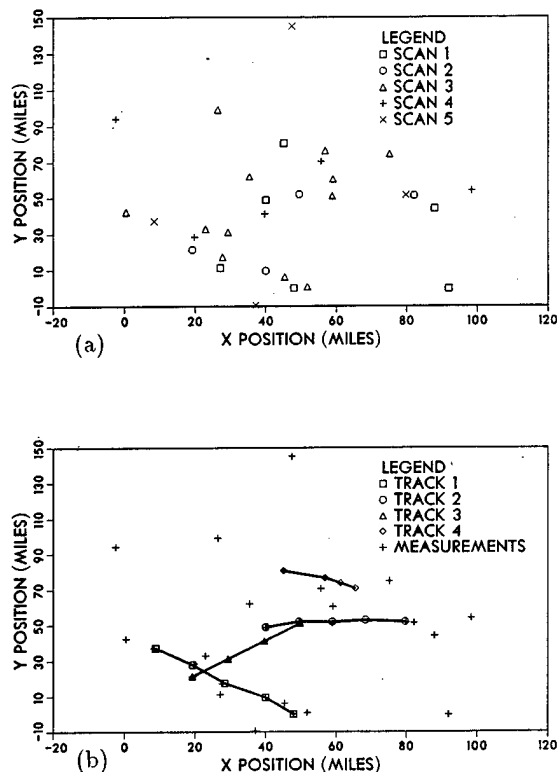


Figure 4. Track Formation for Scenario 2. (a) Received measurements. (b) Initialized tracks.

In the second scenario, all parameters remained the same as in the first scenario except that the probability of detection was lowered to 0.7 and the false alarm rate was increased to 3.5 per scan. For this case, 3 contacts had to be received over the 5 sensor scans before a contact could be considered a candidate track. The received data for this case is shown in Figure 4(a). The bound on the number of targets present was computed to be 5. The initiated tracks are shown in Figure 4(b). Track 4 is a false track, but as seen from the previous scenario, the other tracks

are correct, and two targets were not detected. Note that track 2 was initiated using four measurements and track 3 was initiated with three measurements indicating one and two missed detections in the respective tracks.

VII. CONCLUSIONS

A track initiation algorithm has been presented capable of initiating target tracks in the presence of false alarms and missed detections. The algorithm described is a modification of an algorithm developed by Morefield which addressed multitarget tracking in the framework of pattern recognition. Morefield addressed false alarms but did not address missed detections. It has been shown that by using infinitely erroneous measurements called auxiliary measurements, the multitarget tracking problem can be solved as a set partitioning problem using integer programming so that all measurements are placed in tracks.

A method was described to estimate a bound on the unknown number of targets present in the surveillance area. The purpose of the bound was to improve the efficiency of the algorithm so that the track initiation problem could be solved as a set packing problem using integer programming. Simulation results were presented to demonstrate the algorithm's performance under varying conditions.

Although the algorithm presented can be used to solve the general multi-target tracking problem, its application is limited because of the rapidly growing computational burden associate with it as the number of scans of data increases. Therefore, its main application is to the track initiation problem.

REFERENCES

1. Y. Bar-Shalom, "Extension of the Probabilistic Data Association Filter to Multitarget Environments", *Proc. 5th Symposium on Nonlinear Estimation*, San Diego, CA, September, 1974.
2. Y. Bar-Shalom, T. E. Fortmann, *Tracking and Data Association*, W. F. Ames, ed., Academic Press, Inc., Orlando (1987).
3. S. S. Blackman, *Multiple-Target Tracking with Radar Applications*, Artech House, Inc., Dedham (1986).
4. R. S. Garfinkel, G. L. Nemhauser, *Integer Programming*, John Wiley & Sons, Inc., New York (1972).

5. C. L. Morefield, "Application of 0-1 Integer Programming to Multi-Target Tracking Problems", *IEEE Trans. Automatic Control*, Vol. AC-22, June 1977, pp. 302-312.

6. D. B. Reid, "An Algorithm for Tracking Multiple Targets", *IEEE Trans. Automatic Control*, Vol. AC-24, December 1979, pp. 843-854.

STATISTICAL REASONING SYSTEMS

Dr. Steven S. Noble

Boeing Advanced Systems
P.O. Box 3707, m/s 33-22
Seattle, Washington 98124-2207
Tel. (206) 241-3426

The purpose of this paper is to describe a mathematical system that extends deterministic inferencing mechanisms statistically. The application for Command and Control is to faithfully infer knowledge from statistical, as well as incomplete, data sources. The question of optimizing information storage, while still meeting computational requirements, is also analyzed.

The mathematical structures to be discussed are an outgrowth of classical hypothesis testing, but cover evidential, Bayesian and other reasoning systems. The viewpoint is that we wish to model not the state of an object, but knowledge of the state. As an example, if a fair coin is tossed one can say that the probability of heads is 0.5, but suppose that there is a 50% chance that the coin is fair and a 50% chance that it is some unknown, possibly double-headed, coin. Now what is our knowledge of the state of the coin? Ignoring the question of whether the coin is fair or not we arrive at the following 'conclusion': There is a 25% chance that it is heads, a 25% chance that it is tails, and a 50% chance that we are ignorant of the coin's state. In this process we have constructed a random variable that does not represent the state of the coin, but an inference about the state of the coin.

Since Bayesian as well as evidential reasoning methods are analyzed on an equal footing it is possible to determine the benefit and cost of choosing one over the other. It is also possible to form hybrid schemes that make desirable compromises. For example, a major argument against evidential reasoning methods is the exponential growth of memory and computational resources. The reason this growth occurs is that there is no pruning of hypothetical instances. It is possible, in this analysis, to alter the combination formulas to perform automatic pruning with respect to predetermined rules.

STATISTICAL REASONING SYSTEMS

Dr. Steven S. Noble

Boeing Advanced Systems
P.O. Box 3707, m/s 33-22
Seattle, Washington 98124-2207
Tel. (206) 241-3426

INTRODUCTION

The purpose of this paper is to describe how deterministic inferencing mechanisms can be adapted to data sources which are noisy, incomplete, or inconsistent. While the viewpoint of the approach is based upon evidential reasoning, the constructions are purely probabilistic. The focus will be on modeling the knowledge gained from the data sources. The knowledge will depend on how randomness, or noise, is introduced into a particular data source. We will also address problems that arise in trying to meet computational requirements.

BAYESIAN AND RANDOM INFERENCES

In Bayesian analysis we are given a state space S and a space of observations, or measurements, M . We make a measurement which is, in general, a subset A of M and then compute a distribution on the possible states of the system. To do this we need to have information on how the states and the measurements are correlated. This information is usually expressed in terms of conditional probabilities. That is, for each $s \in S$ there is a probability measure $P_s(\cdot)$ on M , i.e. if A is a subset of M then

$$P_s(A) = \text{Prob}(A|s).$$

In Bayesian analysis the system is described by an *a priori* distribution ρ on S . If we are given this distribution ρ and we have a measurement $A \subset M$, then we can use Bayes formula to compute a new distribution ρ' .

$$\rho'(s) = \text{Prob}(s|A) = \frac{P_s(A)\rho(s)}{\sum_{t \in S} P_t(A)\rho(t)}.$$

Note that the state is considered to be a random variable. In our viewpoint, we will be interested in modeling not the state, but our knowledge of the state as determined by the observation. In this way we may view the state as being deterministic, but unknown. This allows us to dispense with the *a priori* distribution. Consider the classical Bayesian scheme described where for each s in some state space S we have a distribution $P_s(\cdot)$ on a space of observations M . We suppose that each distribution $P_s(\cdot)$ is the distribution of a random variable X_s with values in M . What this gives us is a random map from S to M , that is

$$S \xrightarrow{X} M,$$

where

$$s \mapsto X_s.$$

From the random map X we will construct a map from $P(M)$ to $P(S)$, where $P(B)$ denotes the power set of a set B . This random map will be denoted by X^{-1} and is defined for A a subset of M by

$$X^{-1}(A) = \{s \in S | X_s \in A\}.$$

Note that anytime the conditional distributions $P_s(\cdot)$ are given one can construct this map. The construction depends on the selection of the random variables X_s , which is not unique, but we shall see that there are properties that this map possesses that are independent of the random variables chosen. We shall also see what additional information is contained in the selection.

A question of interpretation arises about the map X^{-1} . By constructing the random variables X_s , we have not only modeled the statistical dependence of the observations on the state, but the actual dependence, the difference being one of knowing the actual random variable verses knowing just its distribution. The additional information allows us to draw conclusions about the state from an observation. Given the observation

$$A \subset M,$$

the conclusion,

$$s \in X^{-1}(A),$$

represents the possible states that could have given that particular observation. In the next section we will abstract this notion, but for now we will think of the results of our 'inferences' as being subsets of S . The randomness in the inference arises from the inherent random quality of the observation.

In order to make coherent decisions based on the outcome of our calculations it is necessary to have a firm probabilistic, or statistical interpretation of the results. We view elements of $P(S)$ as hypotheses about the state of a system. Let h be a hypothesis and let A be an observation, we compute

$$Pl(h) = \text{Prob}(X^{-1}(A) \cap h \neq \emptyset),$$

which reads, the *plausibility* of h . Computing the plausibility in terms of the map X^{-1} we have the

$$\begin{aligned} Pl(h) &= \text{Prob}(X, \epsilon A \text{ for some } s \text{ satisfying } h) \\ &= \text{Prob}(X, \epsilon A | h) \\ &= \text{Prob}(A | h). \end{aligned}$$

Note that comparing the plausibility against a threshold is equivalent to testing the hypothesis after normalizing. The normalized version of the above computation simply requires us to condition on the event that A is observed. This amounts to dividing by $Pl(S) = \text{Prob}(A | S)$. What we then compute is the probability that some $s \in h$ would result in the observation A given that some $s \in S$ would result in the observation A . A brief example will help to clarify the issues involved.

EXAMPLE: THE FAULTY SENSOR I

Suppose that a coin is *not* tossed, but you are relying on some device to detect whether it is heads or tails. Unfortunately this device only works a certain percentage of the time. Let's say that the probability of the device working correctly is p . When it does work it correctly reports the state of the coin, and when it doesn't work the output of the device is independent of the state of the coin. We will suppose that this output can be either heads or tails with equal probability.

If the device informs you that the coin is heads what can we say about the state of the coin? If we compute conditional probabilities we find

$$\begin{aligned} \text{Prob}(\text{device says heads} | \text{heads}) &= (1+p)/2 \\ \text{Prob}(\text{device says heads} | \text{tails}) &= (1-p)/2. \end{aligned}$$

To use Bayes formula we have to make an assumption about the *a priori* state of the coin. We assume that the state of the coin is random and that either heads or tails are equally likely. If we do so, then using Bayes formula we find

$$\begin{aligned} \text{Prob}(\text{Heads}) &= (1+p)/2 \\ \text{Prob}(\text{Tails}) &= (1-p)/2. \end{aligned}$$

That is, the state of the coin is still random, only the probability distribution has been changed.

On the other hand if didn't assume that the state of the coin is random and if we use the above construction to model the random inference we find that we have a set valued random variable $A = X^{-1}(\text{heads})$ such that

$$\begin{aligned} \text{Prob}(A = \{\text{heads}, \text{tails}\}) &= (1-p)/2 \\ \text{Prob}(A = \{\text{heads}\}) &= p \\ \text{Prob}(A = \emptyset) &= (1-p)/2. \end{aligned}$$

If we condition off the empty set we have

$$\begin{aligned} \text{Prob}(A = \{\text{heads}, \text{tails}\}) &= (1-p)/(1+p) \\ \text{Prob}(A = \{\text{heads}\}) &= 2p/(1+p). \end{aligned}$$

In the first calculation the state of the coin is random and in the second, what we infer about the coin state is random. We can now easily test the hypothesis that the coin is heads

$$Pl(\text{heads}) = 1$$

which means that we have not received any information to decrease the likelihood of heads, while testing the hypothesis of tails results in

$$Pl(\text{tails}) = (1-p)/(1+p).$$

There are two things to note here. The first is that both methods give the same results as far as this standard statistical test. The second is that by studying the inference mechanism involved we have a natural language for handling the logical relationships between hypotheses. In this example, as it turns out, we have more accurately modeled the randomness of the experiment by the second approach. Now what happens when we assume the existence of an *a priori* distribution?

Suppose that Z is an S valued random variable such that

$$\text{Prob}(Z \in B) = \rho(B).$$

We can think of Z as taking values in the singleton sets. To avoid confusion, define $\hat{Z} = \{Z\}$. Given \hat{Z} and X^{-1} , we can logically combine them. Define a map Y from $P(M)$ to $P(S)$ by

$$Y(A) = X^{-1}(A) \cap \hat{Z}.$$

Note that Y takes values on the singleton sets and the empty set. If we condition $Y(A)$ off the empty set then it can be thought of as an S valued random variable. If we then compute the distribution of this random variable we arrive at Bayes formula.

$$\begin{aligned} \rho'(s) &= \text{Prob}(Y(A) = \{s\} | Y(A) \neq \emptyset) \\ &= \text{Prob}(Y(A) = \{s\}) / \text{Prob}(Y(A) \neq \emptyset) \\ &= \frac{\text{Prob}(\{\hat{Z}\} \cap X^{-1}(A) = \{s\})}{\text{Prob}(\{\hat{Z}\} \cap X^{-1}(A) \neq \emptyset)} \\ &= \frac{\text{Prob}(Z = s, Z \in X^{-1}(A))}{\text{Prob}(Z \in X^{-1}(A))} \\ &= \text{Prob}(Z = s, X_Z \in A) / \text{Prob}(X_Z \in A) \\ &= \text{Prob}(s | A) \end{aligned}$$

This calculation is independent of the choice of the random variables X . This calculation shows that, if desired, we can still perform Bayesian analysis.

In the absence of an *a priori* distribution we could assign masses to sets as in evidential reasoning. For BCS we define

$$m_A(B) = \text{Prob}(X^{-1}(A) = B).$$

Note that this mass depends explicitly on the observation, or evidence A . The above equation represents unnormalized mass i.e. there is a potential nonzero mass assigned to the empty set. This mass is

$$m_A(\phi) = \text{Prob}(X^{-1}(A) = \phi) \\ = \text{Prob}(X_s \in A^c \forall s \in S),$$

where A^c is the complement of A in M . The corresponding normalized mass is given by conditioning on $X^{-1} \neq \phi$. For example, the mass on B given $X^{-1}(A)$ is not empty is

$$\bar{m}(B) = P(X^{-1}(A) = B | X^{-1}(A) \neq \phi) \\ = \frac{\text{Prob}(X^{-1}(A) = B; X^{-1}(A) \neq \phi)}{\text{Prob}(X^{-1}(A) \neq \phi)} \\ = \frac{\text{Prob}(X^{-1}(A) = B)}{(1 - \text{Prob}(X^{-1}(A) = \phi))} \\ = m(B)/(1 - m(\phi))$$

If we perform an independent experiment then this gives rise to an independent collection of random variables \bar{X} , parametrized by S and with values in \bar{M} , where \bar{M} is a new space of observations (perhaps equal to M). By independent, we mean that for each $s' \in S$, $\bar{X}_{s'}$ is independent of the collection of random variables $\{X_s, s \in S\}$ and similarly for \bar{X} and X reversed. From the collection \bar{X} , we construct \bar{X}^{-1} from $P(\bar{M})$ to $P(S)$ and the corresponding mass function $\bar{m}_{\bar{X}}$ for each $\bar{A} \in P(\bar{M})$. We can now logically combine X^{-1} and \bar{X}^{-1} , defining the map Y from $P(M) \times P(\bar{M})$ to $P(S)$ by

$$Y(A, \bar{A}) = X^{-1}(A) \cap \bar{X}^{-1}(\bar{A})$$

If we compute the corresponding mass function for Y we get

$$m_{A \times \bar{A}}(B) \\ = \text{Prob}(Y(A, \bar{A}) = B) \\ = \text{Prob}(X^{-1}(A) \cap \bar{X}^{-1}(\bar{A}) = B) \\ = \sum_{C \cap D = B} \text{Prob}(X^{-1}(A) = C, \bar{X}^{-1}(\bar{A}) = D) \\ = \sum_{C \cap D = B} \text{Prob}(X^{-1}(A) = C) \text{Prob}(\bar{X}^{-1}(\bar{A}) = D) \\ = \sum_{C \cap D = B} m_A(C) \bar{m}_{\bar{A}}(D)$$

This is the unnormalized version of Dempster's rule of combination. To normalize we condition Y off the empty set. This amounts to dividing by

$$1 - m_{A \times \bar{A}}(\phi).$$

Note that the only difference between Bayesian reasoning and evidential reasoning

is the type of inferences that were combined. In Bayesian reasoning we combined the inference with an inference derived from an *a priori* distribution. In evidential reasoning we combined inferences derived from two independent observations.

It is also possible to compute mass distributions on sets from the conditional probabilities found in the purely Bayesian approach. If we assume that the random variables X_s are independent then the normalized mass on a set B in S given a measurement A in M is

$$m_A(B) = \frac{\prod_{s \in B} P_s(A) \prod_{s \in B^c} (1 - P_s(A))}{(1 - \prod_{s \in S} (1 - P_s(A)))}.$$

If we then combined this distribution with the mass distribution derived from an *a priori* distribution using Dempsters rule we would have Bayes formula.

In the next section we will study properties of inferences a little more closely and we shall consider a variety of combination formulas that will narrow the gap between evidential and Bayesian reasoning further.

INFERENCE MAPS

We wish now to focus our attention on X^{-1} . This is a random map from the power set $P(M)$ to the power set $P(S)$. The natural generalization of a power set is a lattice. Briefly, a lattice L is a partially ordered set with two operations, \wedge and \vee , with the property that

$$a \wedge b = \inf\{a, b\} \\ a \vee b = \sup\{a, b\}$$

(\sup = least upper bound and \inf = greatest lower bound). Since the lattices encountered here will be finite, we can extend this operation to arbitrary subsets. For example, if L' is a subset of L then

$$\sup(L') = \bigvee_{l \in L'} l$$

$$\inf(L') = \bigwedge_{l \in L'} l$$

both exist in L . In particular, we define θ to be the $\sup(L)$ and ϕ to be the $\inf(L)$. Note that the power set, $P(S)$, can be thought of as a lattice of propositions about an element $s \in S$, where the proposition corresponding to a subset B of S is ' $s \in B$ '. The partial order is given by set inclusion, \wedge is the operation of intersection, and \vee is the operation of union.

Definition 1 Let L_1 and L_2 be Lattices and let I be a map from L_1 to L_2 . We will say that I is an inference if I is order preserving.

An inference therefore preserves the logical implication of statements. The map X^{-1} from the previous section is an example of an inference from the lattice of observations $P(M)$ to the lattice $P(S)$. We also constructed, from an *a priori* distribution, a random variable \hat{Z} taking values in the lattice $P(S)$. We may extend this to a random inference \hat{Z} defined on the Boolean lattice $B = \{0, 1\}$ by

$$\hat{Z}(1) = \hat{Z} \\ \hat{Z}(0) = \phi$$

Suppose that

$$L_1 \xrightarrow{I} L_2$$

is a random inference, we define for $e \in L_1$ and $h \in L_2$

$$m_e^I(h) = \text{Prob}(I(e) = h)$$

the distribution of I . Suppose that $L_1 \xrightarrow{I_1} L$ and $L_2 \xrightarrow{I_2} L$ are two inference maps. We can combine I_1 and I_2 by taking either the conjunction or the disjunction of I_1 and I_2 . We first form the product lattice $L_1 \times L_2$ and define the maps

$$L_1 \times L_2 \xrightarrow{I_1 \wedge I_2} L,$$

and

$$L_1 \times L_2 \xrightarrow{I_1 \vee I_2} L,$$

by

$$\begin{aligned} I_1 \wedge I_2(e_1, e_2) &= I_1(e_1) \wedge I_2(e_2) \\ I_1 \vee I_2(e_1, e_2) &= I_1(e_1) \vee I_2(e_2). \end{aligned}$$

There is also a distribution version of these formulas. If we assume that I_1 and I_2 are independent then

$$m_{e_1, e_2}^{I_1 \times I_2}(h) = \sum_{h_1 \wedge h_2 = h} m_{e_1}^{I_1}(h_1) m_{e_2}^{I_2}(h_2)$$

To normalize the equation we condition off the event $I_1 \wedge I_2(e_1, e_2) = \phi$. If we set

$$k = \sum_{h_1 \wedge h_2 = \phi} m_{e_1}^{I_1}(h_1) m_{e_2}^{I_2}(h_2)$$

then we have the normalized combination formula.

$$\bar{m}_{e_1, e_2}^{I_1 \times I_2}(h) = \left(\frac{1}{1-k} \right) \sum_{h_1 \wedge h_2 = h} m_{e_1}^{I_1}(h_1) m_{e_2}^{I_2}(h_2) \quad (1)$$

Referring to the previous section, if $L_1 = P(M)$, $L_2 = B$, $I_1 = X^{-1}$, and $I_2 = \hat{Z}$ then (1) is Bayes formula. For $L_1 = P(M)$, $L_2 = P(\bar{M})$, $I_1 = X^{-1}$, and

$I_2 = \bar{X}^{-1}$ then (1) is Dempster's rule of combination. Equation (1) can of course be extended to multiple sources or inferences. Since it is the distributional version of a logical combination, all logical properties are preserved by the distribution. In this way one gets the associative law.

Suppose that $L_1 \xrightarrow{I_1} L$ and $L_1 \xrightarrow{I_2} L$ are two inference maps defined on the same lattice L_1 . For $0 \leq \lambda \leq 1$ we define the mixture

$$I = \lambda I_1 + (1-\lambda) I_2$$

by

$$I(e) = \begin{cases} I_1(e) & \text{if } U \leq \lambda \\ I_2(e) & \text{if } U > \lambda \end{cases}$$

where U is an independent, uniform $[0,1]$ random variable. Notice that the distribution of the mixture is the mixture of the distributions,

By taking various combinations of mixtures, conjunctions, and disjunctions, we get a plethora of combination formulas. For instance, given $L_1 \xrightarrow{I} L$ and $L_2 \xrightarrow{J} L$, we can define

$$Q = \lambda I \wedge J + (1-\lambda) I \vee J$$

The interpretation of Q is that it lies somewhere between choosing to consider both I and J are correct and at least one of I or J is correct.

We can also define

$$I^\lambda = \lambda I + (1-\lambda) \hat{\theta}$$

where $\hat{\theta}$ from L_1 to L is constantly equal to θ . I^λ mixes I with a trivial inference.

The reason for considering lattices is to allow for more general structures that do not require as many hypothesis and consequently have a reduced computational load. For example consider the lattice of subsets of a set S , L , consisting of the empty set, the set S , and the singleton sets and define the following inference from $P(S)$ to L

$$J(B) = \begin{cases} B & \text{if } B \text{ is empty or a singleton} \\ S & \text{otherwise} \end{cases}$$

There is an obvious reduction in the number of hypothesis to consider, but with a corresponding penalty. Information about B is lost unless B is a singleton. Although this example is extreme it is useful when we are making measurements that directly correspond to states and we have a minimal amount of statistical information. The is illustrated in the next example.

EXAMPLE: THE FAULTY SENSOR II

Let $S = \{s_1, \dots, s_N\}$ and let the measurement space M equal S . Suppose that we have a sensor that is to determine the state of the system, that is, it returns values s_1, \dots, s_N . We make the assumption that with probability p the sensor accurately assesses the state and is otherwise uncorrelated with the actual state.

Under these conditions we see that as a lattice the measurement space as well as the hypothesis space can be represented by L . We define now the random inference map from L to itself as follows

$$\begin{aligned} J(S) &\equiv S \\ J(\{s\}) &= \begin{cases} \{s\} & \text{if the sensor is accurate} \\ S & \text{otherwise} \end{cases} \\ J(\phi) &\equiv \phi. \end{aligned}$$

Of course we do not know whether the sensor is working or not, but we do know the probability,

$$\begin{aligned} \text{Prob}(J(\{s\}) = \{s\}) &= p \\ \text{Prob}(J(\{s\}) = S) &= 1-p. \end{aligned}$$

Notice that $J = I^p$ in distribution where $I : L \rightarrow L$ is the identity map. We can use our update formula (1) to combine two independent measurements. An advantage of this approach is that a single number is used to measure confidence in the sensor.

In order to determine what value we should assign to p , we can think of performing a series of independent experiments. The results of the experiments will be a series of states S_1, S_2, \dots . Assume that the system is in state s_k , and there exists, unbeknownst to us, conditional probabilities p_1, p_2, \dots, p_N , $p_j = \text{Prob}(S_n = s_j)$. If we compute the ratios of the plausibilities after n of experiments we find

$$\frac{\text{Pl}(s_k)}{\text{Pl}(s_i)} = (1-p)^{N_k - N_i}$$

where N_j is the number of times that the sensor reported state s_j . if

$$0 < p < 1 - \alpha$$

where $\alpha = \sup_{j \neq k} (p_j/p_k)$, then $\text{Pl}(s_j)/\text{Pl}(s_k) \rightarrow 0$ for $j \neq k$. if

$$p = 1 - \sqrt{\alpha}$$

then we have optimized the rate of convergence in the sense of expected values. If one wishes to determine how many samples must be taken in order to ensure that

$$\text{Prob}\left(\frac{\text{Pl}(s_k)}{\text{Pl}(s_j)} < T \forall j \neq k\right) > 1 - \epsilon$$

where ϵ is a small number, then, using the above value of p , we must have

$$n \geq \log(\epsilon T / (1 - \beta))$$

where $\beta = \sup_{j \neq k} P_j + p_k - 2\sqrt{p_j p_k}$

To compare this with a purely Bayesian inference, we suppose that we have estimated

$$\text{Prob}(S_j = s_k | s_k) = p$$

with the rest of the mass being distributed over the remaining states equally. In this case, in order to have convergence in mean, we must have

$$\frac{1}{N} < p < \frac{1}{\alpha(N-1)+1},$$

with the optimal value

$$p = \frac{1}{\sqrt{\alpha(N-1)+1}}$$

We see that choosing this probability for the purely Bayesian inference depends intimately on the number of states in question. In addition, there is a narrower range of acceptable values that may be used. For example if $N = 100$ and $\alpha = .5$, which means that the state s_k is twice as likely as any other state, then from the Bayesian viewpoint we must have

$$0.01 < p < 0.02,$$

while for our inference we only need

$$0.0 < p < 0.5$$

to have convergence. The optimal values are 0.014 for the Bayesian inference and 0.3 for our first inference.

In addition, one still has the option in either case to combine the result with an *a priori* distribution, as represented by an inference Z from the boolean lattice to L . It is also possible to combine with Z' , for instance, if the *a priori* distribution is a statistic based on only a percentage of the population.

In general, if $L \subset P(S)$ is a collection of subsets closed under intersection, then we can turn L into a lattice together with an inference $J: P(S) \rightarrow L$. This will allow the tailoring of the lattice to the desired hypotheses that need to be tested. In order to compute the distribution of the combined inference from $P(M)$ to L we need to model the information that is contained in a measurement with respect to L .

SUMMARY

In the first section it was shown that by considering inferences one can refer to either Bayesian or evidential reasoning, that it is possible to combine Bayesian and evidential reasoning in the sense that starting with partial Bayesian information, i.e. conditional probabilities, one can construct mass functions on sets as in Dempster-Shafer theory, and that the mass distribution when combined with an *a priori* distribution, was in agreement with Bayes formula. It was also shown that the desired update formulas could be interpreted as logical combinations of

inferences and that a consistent statistical measure can be applied to interpret the results.

In the second section we abstracted the notion of an inference from subsets to lattices and developed a corresponding combination formula. By considering lattices we are freed from having to construct a mass distribution on the full power set and an example was given that had a minimal number of hypotheses. Finally we compared this example to the corresponding Bayesian situation.

One question that comes up is that of the normalization. This has a ready interpretation since the normalization factor is $Pl(\theta)$. Logically this means that we are conditioning off of 'False'. We do this if we believe that our assumptions are correct probabilistically. If $Pl(\theta)$ is below some threshold, then we are conditioning on an event of low probability; this is a signal that we are combining erroneous data. Thus, the normalization factor serves as an error warning. This can also serve as a measure of agreement when combining data from many sources

It is also possible to measure the amount of statistical information and at various levels. $\text{Prob}(I=\theta)$ measures, in a sense, the total ignorance of the inference mechanism. Note, that, using this definition, a Bayesian system has complete statistical information when one uses the *a priori* distribution.

It is hoped that the observations in this paper will help to quell the levels of controversy of whether one should perform Bayesian verses evidential reasoning, in that they can both be considered as inference mechanisms. The important question is how to statistically model the information in a data source. A few general examples were given,

but they are foundational in nature. Specific examples require more in depth study, such as the generalized tracking problem where the system evolves with time as measurements are taken. It is possible to treat the evolution of states also as an inference, an inference that has less statistical information as time increases, i.e.

$$\text{Prob}(I_t = \theta)$$

increases with time.

As a final observation, we note that the space of fuzzy sets is also a lattice. Of course, now it is no longer finite so that the problem of working over a continuous lattice is presented. Although the results are not conclusive, preliminary work indicates that in order to work with infinite lattices, additional structure is desired; for example, we may require our inference to satisfy a continuity property. This is an as yet unexplored area that awaits investigation.

ALGORITHM FOR DECISIONS ON MERGING AND LINKING TARGET TRACKS

J. B. Campbell

on leave,
UNISYS

J. E. Samaan

Staff Systems Engineer
IBM, Crystal City, VA
1-703-841-7460

Abstract

Definitions are reviewed of report-to-report and track-to-track correlation with respect to target tracks in surveillance systems. The need for track-to-track correlation in merging and linking tracks is indicated. An algorithm for automatic decision making in merging and linking tracks is derived from fundamentals of target location information in reports received by surveillance systems. The algorithm is applied successfully to data that are representative of the systems. The illustrative examples demonstrate algorithm usefulness not only for track-to-track correlation, but also for report-to-track correlation.

ALGORITHM FOR DECISIONS ON MERGING AND LINKING TARGET TRACKS

J. B. Campbell

on leave,
UNISYS

J. E. Samaan

Staff Systems Engineer
IBM, Crystal City, VA
1-703-841-7460

1. INTRODUCTION

The function of some surveillance systems is the representation of target tracks derived from reports received from various sources. The reports contain two kinds of information about the targets: (a) location of a target at a specific time; and (b) non-local information, such as the vessel type, identifying physical attributes, or electronic emission characteristics.

Surveillance systems algorithms use both kinds of information to sort reports by target, and to string the locations together to represent the track of the target. The process of assigning the information of a report to a track is known as "report-to-track correlation". Surveys of such correlation algorithms are given in reports published by the Naval Research Laboratory. [1,2]

Report-to-track correlation can assign a report to one track only, but cannot guarantee that a platform is associated with just one track given a sequence of reports. In fact, the known report-to-track correlation algorithms will frequently yield a multiplicity of tracks per platform. That condition can occur when a platform is being tracked by more than one sensor. It can occur also when an emitter-tracked platform activates more than one emitter, or operates an emitter in different modes.

An algorithm due to Reid [3] and an extension of that algorithm to process data from multiple sensors [4] have been successful in decreasing the multiple track representation of single tracks by a technique for fusing data on the same target received from multiple sensors. However, the technique cannot remove all ambiguities, particularly when there are large time gaps in the sequence of reports, and when reports arrive

out of sequence. Preliminary tests show that the technique can segment a single track, representing the segments as different tracks.

"Track-to-track correlation" is required to reduce the multiple track representation of a platform to single track representation.

This paper presents a method for linking the segments of the same track, and for merging multiple representations of a single track. The paper derives a track-to-track correlation algorithm from (a) the ergodicity of the process formed by measurements of a moving target taken by different sensors and (b) the change induced is a best estimate of the track when one set of measurements is merged with another set that may or may not be attributable to the same target.

The successful application of the algorithm to track merging and linking shows the potential of the method as a much needed supplement to current tracking techniques. Weighted, least squares splines, referenced in the paper, are used as readily implementable low-pass filters.

A formal statement of the problem, and a derivation of the algorithm as a solution follows.

2. PROBLEM

Statement of the Problem

From non-local information about two targets, each associated with a distinct track formed by measurements of different sensors, and from the times of the measurements, it is known that the targets may not be different. Assume that the times are error-free, and that for the position data, uncorrupted by systematic error, there are

estimates of the measurement error. Find a measure of correlation between the targets.

Discussion

Track of Target. A track is a collection of geographic points that represent positions of a moving platform, known as a target. Sensor measurements of positions form a probability distribution, the first moment of which is a function of time. If sensor measurements are free of systematic error, the first moment is a best estimate of the track.

When measurements of a target made by two or more sensors are combined, the resultant combination of distributions forms a stochastic process. If every sensor contributing to the process is free of systematic error, and every sensor is measuring the same target, the process must be ergodic. That is, the first moment of a distribution must represent "the same" track as the first moment of the ensemble. And if the distribution corresponding to one target is combined with that corresponding to a different target. The first moment of one distribution and that of the ensemble represent tracks that are "not the same".

Practical definitions of "the same" and of "not the same" require analysis of a relatively large body of empirical data. Pending such an analysis it is possible to set tentative definitions by analogy with performance specifications for communication filters, or by analogy with an acceptable finite series approximation to a function. Noting that in this case the first moments are deterministic functions, the first moment of a distribution and the first moment of an ensemble may be said to be "the same" if the functions are within 3 dB of each other, and "not the same" if they are at least 6 dB apart.

Model of Track. The actual track is of a form that can be modeled by a mathematical expression suitable for best estimate techniques. For example, the track can be modeled by a velocity vector or by means of polynomials fitted to intervals in a least squares sense. A velocity vector model has been customary in tracking with the aid of Kalman type filtering. Justification for a spline model follows.

Weighted Least Squares Spline. Consider that the targets being tracked can change direction no faster than aircraft. The

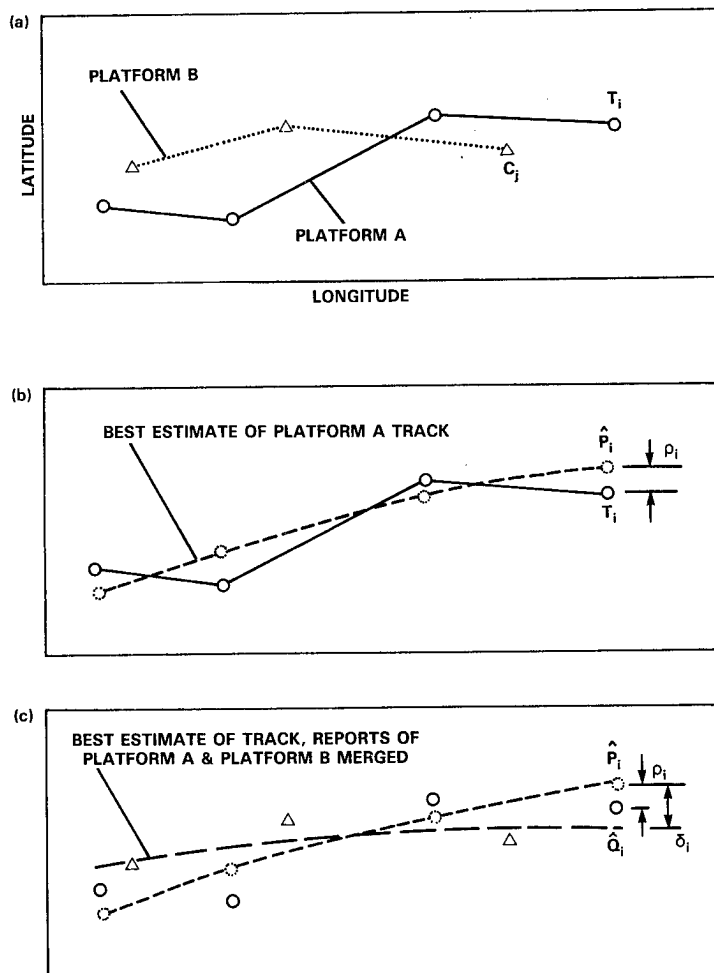


Figure 1

Large Displacement from Best Estimate
Induced by Reports from Another Platform

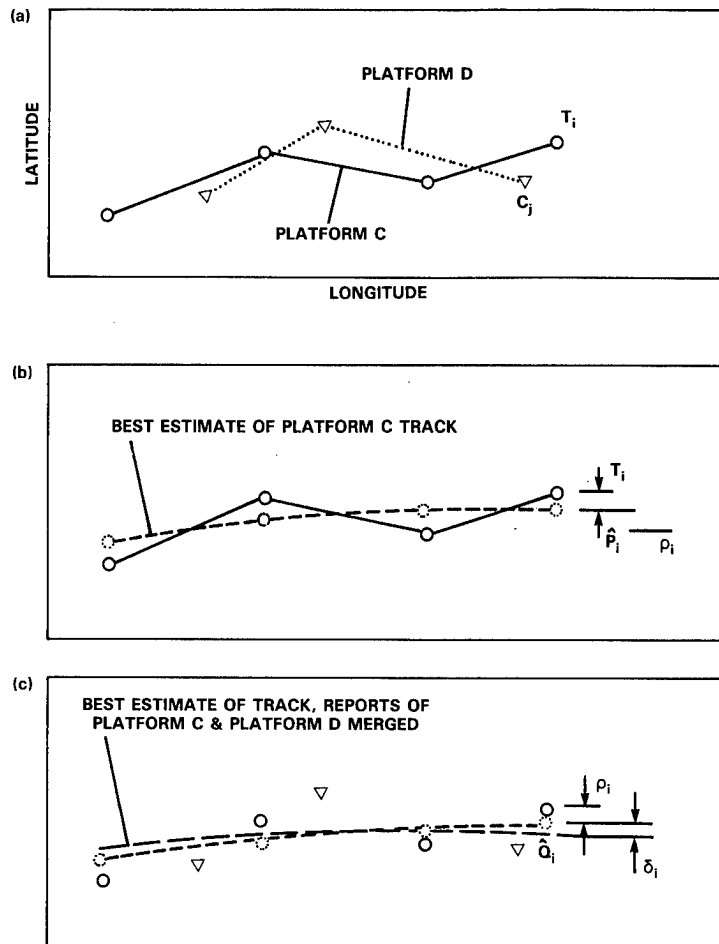


Figure 2

Small Displacement from Best Estimate
Induced by Reports from Same Platform

tracks reconstructed by the surveillance system are composed of sampled target locations. Say that the spacing of samples is such that they include at least one point of a maneuver. Then the tracks are piecewise representable by polynomials of degree no higher than three. That is, the report positions can be used as the knots of a cubic spline. Where an estimate of the measurement error accompanies the reported location, a weighted least squares cubic spline [5] can be used to represent the target track in segments, with the endpoints of a segment coinciding with a pair of adjacent knots. There is a weighted, least squares spline representation [6] such that a segment is representable by a closest fit polynomial, of first, second or third degree, whichever is most applicable. This representation is equivalent to a smoothing or low-pass filter. [7]

Best Estimates, Target Location, Location Error. From either the Kalman type filtering or the weighted, least squares spline fit to the position data, there are obtainable:

(a) A best estimate of the true location at the time of the measured location, and

(b) A best estimate of the error in the measured location. Both are useful in computing the correlation of measurements.

Displacement of Best Estimate Track. Consider two sets of reports, each possibly associated with the same target. Call one set the Tester, the other set, the Candidate. The Tester set is one distribution of measurements. Find the first moment, making a best estimate of the Tester track, and note the estimated errors (residuals). Merge the Tester and the Candidate sets of reports, forming a process. Call the result the Mergence. Find the first moment of the Mergence, making a best estimate of the Mergence track.

The best estimate Mergence track contains a subset of positions that correspond to the positions of the original best estimate Tester track. Call this subset the Displaced Tester.

The correlation of Candidate to Tester can be determined by comparing the energy of the function representing the Tester with that of the function representing the Displaced Tester. Alternatively, the correlation can be found from the displacement of the Tester.

For the alternative technique, find the distance between each point of the Tester and the corresponding point of the Displaced Tester. This displacement in the original best estimate Tester track is induced by the Candidate track. If the displacement is negligible, the process can be considered ergodic; or Tester and Candidate tracks judged to be due to the same target. If the displacement is large, Tester and Candidate tracks most likely were due to different targets.

Displacement Illustrations. Figures 1 and 2 illustrate the displacement principle. In the illustrations target positions are in geographic coordinates, with time of locations and values of longitudes error-free. In Figure 1, Target A is different from Target B. In Figure 2, Targets C and D are the same. In each illustration, the two tracks are due to independent sources of observations.

Figure 1b shows the measured locations and the corresponding best estimate locations of the track for Target A, the Tester. Figure 1c compares the Tester with the best estimate obtained when observations from Targets A and B are merged. The displacement is noticeably large, indicating that Target A is different from Target B.

A similar comparison is shown in Figures 2b and 2c. Here the induced displacement of the Tester track positions is minor, indicating that Target C is the same as Target D.

Displacement, Measurement Error, and Correlation. The displacement is error in the Tester track disclosed by or resulting from the merging of Candidate and Tester. The total error in the Tester track after the merging is the sum of the measurement error and the displacement. Comparison of Tester track measurement error with the total error in the track after merging can be made equivalent to a comparison of the original Tester track with the displaced track for the purpose of computing correlation. When the track is on a curved surface, it is convenient to compute the measure of correlation in terms of the values of the errors, which are generally much smaller than the values of the track coordinates.

3. MEASURE OF CORRELATION

Tester Error

At some time τ_i , let T_i , the measured location of a moving target be accompanied by an estimate of the uncertainty in the measurement.

$$T_i = P_i + \epsilon_i$$

where P_i is the true location, ϵ_i is the error in the measurement of time τ_i , and all three terms are vectors. If there is no systematic error, the mean of the measurement error is 0. Then the mean of T_i equals P_i .

Collect a set of n measurements, $\{T_i, i=1,2,\dots,n\}$. Find \hat{P}_i , the best estimate of P_i ($i=1,2,\dots,n$). The best estimate of the error is the residual ρ_i .

$$\rho_i = \hat{P}_i - T_i$$

Change Induced by Candidate

At some time τ_i , let the measured location of a moving target made by another sensor by C_j , accompanied by an estimate of the uncertainty in the measurement. Collect a set of m measurements $\{C_j, j=1,2,\dots,m\}$ and add to the $\{T_i\}$ set. Find the best estimate of the track represented by $\{T_i + C_j\}$, and select the points $\{\hat{Q}_i, i=1,2,\dots,n\}$ that correspond to the points $\{T_i\}$. \hat{Q}_i is a new best estimate of P_i . The new best estimate is displaced from the old by δ_i .

$$\delta_i = \hat{Q}_i - \hat{P}_i$$

Measure in Terms of Best Estimate Tracks

The energy of the first best estimate of $\{P_i\}$ is

$$E_1 = \hat{P}_i^2$$

that of the new best estimate,

$$E_2 = \hat{Q}_i^2$$

Formally, the measure of correlation can be written as M^2 ,

$$M^2 = E_1/E_2$$

The estimates are within 3 dB of each other when $M^2 \geq 0.5$, and are at least 6 dB apart when $M^2 \leq 0.25$.

Where target location coordinates are in earth latitude and longitude, calculation of the terms composing E_1 and E_2 depends on the earth model used. The simplest earth model to use is a sphere. But where the track extends over hundreds of miles, the spherical earth model can introduce significant error in the calculations. More accurate earth models complicate the calculations considerably.

Relative to the latitude/longitude components of target location, residuals and displacements are small enough for their energies to be calculated with the use of a spherical earth model. Therefore, it is convenient to write the measure of correlation in terms of residuals and displacements.

Let R be the root-mean-square (rms) value of the set of residuals $\{\epsilon_i, i=1,2,\dots,n\}$, and D , the rms value of the set of displacements $\{\delta_i, i=1,2,\dots,n\}$. R^2 and D^2 are defined by

$$R^2 = (1/n) \sum_{i=1}^n (\epsilon_i)^2 \text{ and}$$

$$D^2 = (1/n) \sum_{i=1}^n (\delta_i)^2$$

$R + D$ is the estimated total error in the Tester after the merge.

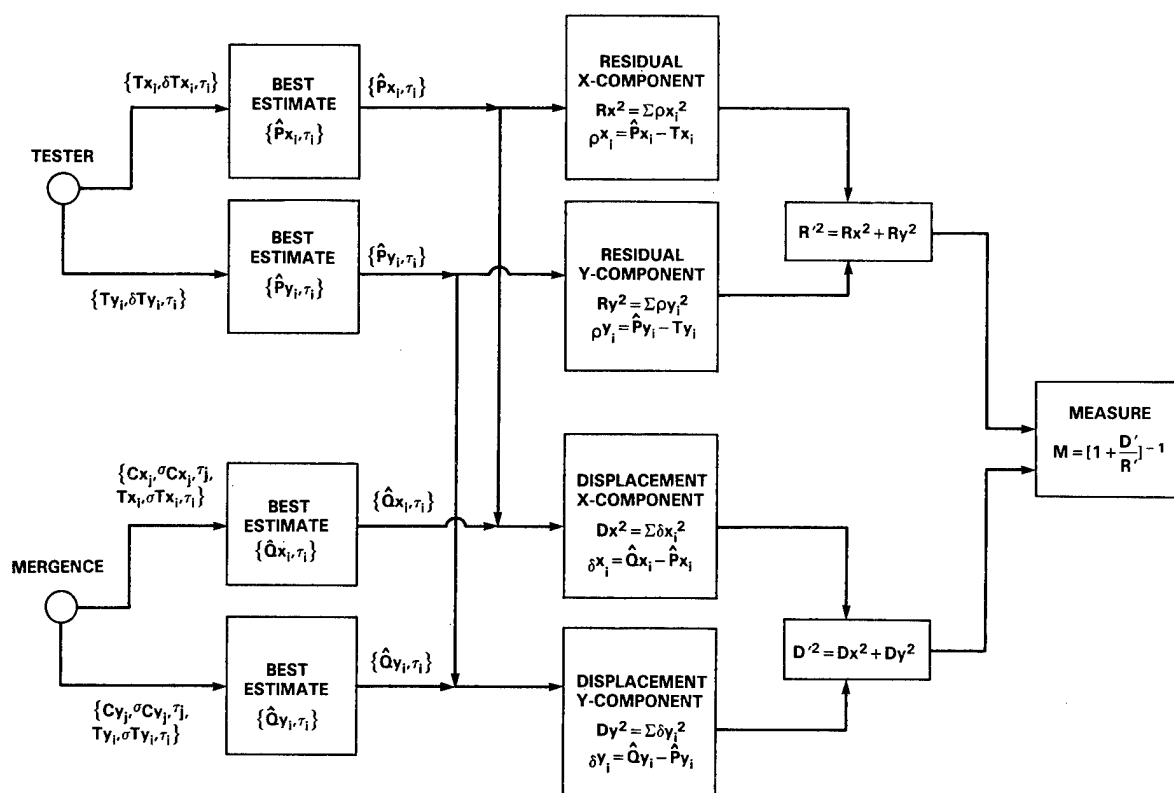


Figure 3

Calculation of Measure of Correlation for
2-Dimensional Cartesian Coordinates

The measurement error relative to the total error is M,

$$\begin{aligned}
 M &= R/(R + D) \\
 &= 1/[1 + (D/R)] \\
 &= [1 + (D'/R)]^{-1}
 \end{aligned}$$

where $D'^2 = nD^2$, and $R'^2 = nR^2$. That is, the root-sum-square (rss) values of residuals and displacements can be substituted for the rms values in the calculation of M.

Note that M (which compares the measurement error of the Tester track function, with the total error of that track function after the merge of Tester with Candidate track) varies directly with the square root of the ratio of the energy of the two functions. Pending empirically determined thresholds, it can be expected that the functions are the same, or that Tester and Candidate tracks belong to the same target when $M \leq 3$ dB; and that they belong to different targets when $M \geq 6$ dB. Equivalently,

if $M \geq 0.7$, accept correlation of the targets,

if $M \leq 0.5$, reject correlation of the targets.

If $0.5 < M < 0.7$, correlation of the targets is uncertain.

4. CALCULATION OF MEASURE

Cartesian Coordinates

Calculation of the measure using rss error is diagrammed in Figure 3 for locations given in two-dimensional Cartesian coordinates. The extension to three dimensions is obvious.

The best estimate calculation is effected in stages, one for each dimension of the location. Each dimension is accompanied with the appropriate component of uncertainty. For example, the x-component Tx_i is accompanied with σTx_i .

Geographic Coordinates

The calculation in stages is applicable to locations of targets on the surface of the earth. Each location is given as a latitude-longitude pair together with an uncertainty ellipse defined by semi-major and semi-minor axes in nautical miles, plus orientation of the major axis relative to North. Allocation of uncertainty to latitude and to longitude is obtainable from the ellipse parameters by techniques for coordinate system change. The residual (or displacement) error can be calculated as the hypotenuse of the spherical right triangle with legs equal to the residuals (or displacements) in latitude and longitude.

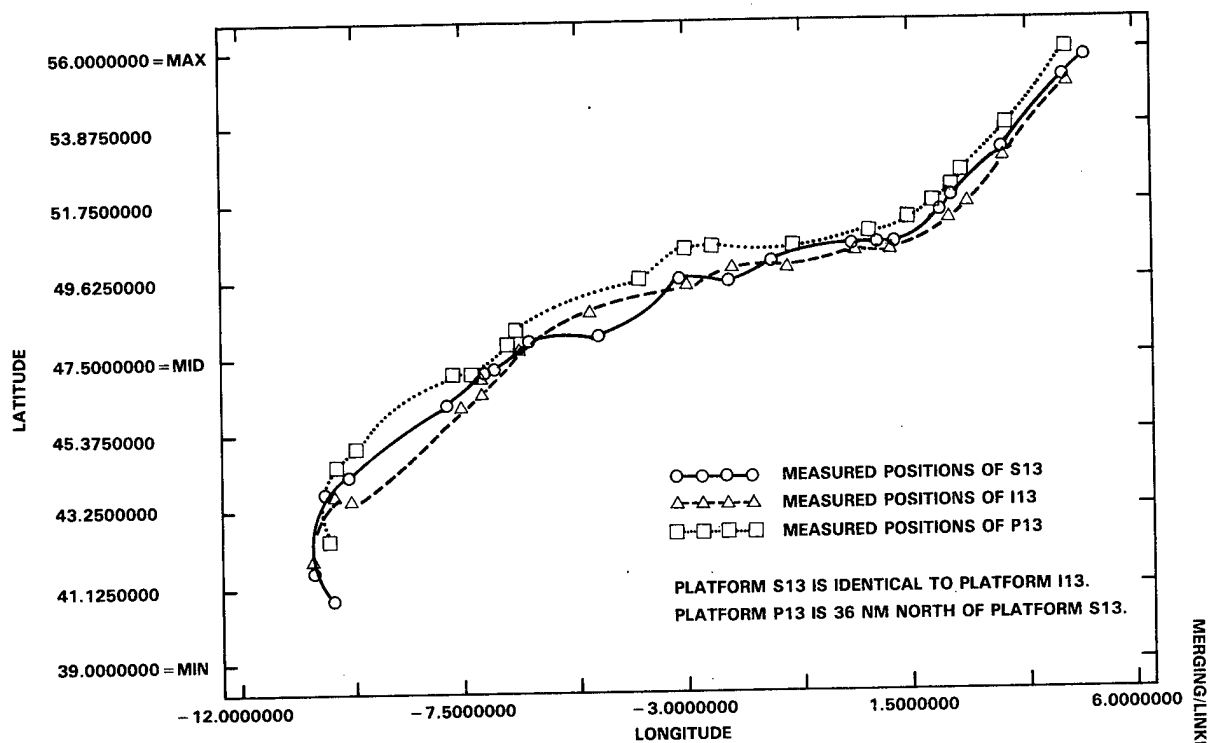


Figure 4

Plot of Three Closely Spaced Tracks

TABLE I
Locations for Simulated Target S13

	Time D/H/M	Location		Uncertainty		
		Lat	Lon	a	b	θ
A	0/14/55	4030	-1002	1	0	87
	0/19/29	4139	-1026	12	6	74
	1/04/06	4343	-1007	1	0	12
	1/06/44	4414	-0937	3	0	52
	1/16/07	4606	-0739	10	5	20
	1/19/52	4701	-0701	12	8	28
	1/20/39	4709	-0650	1	0	4
	2/00/21	4754	-0603	2	2	37
B	2/05/55	4802	-0433	94	38	58
	2/11/45	4945	-0255	4	1	27
	2/15/24	4943	-0206	7	1	5
	2/19/53	5003	-0109	2	1	32
	3/03/20	5024	00029	24	7	0
C	3/05/11	5021	00054	1	0	30
	3/06/09	5028	00057	8	1	1
D	3/09/04	5040	00115	2	1	13
	3/17/22	5118	00212	3	2	39
	3/18/23	5127	00215	13	11	53
E	3/20/58	5150	00229	1	0	98
	4/05/27	5309	00325	1	0	65
	4/18/30	5509	00446	1	0	2
	4/22/09	5543	00509	3	2	59

Time: D/H/M = Days/Hours/Minutes lapsed from zero time
 Location: Latitude, Longitude in Degrees and Minutes
 Uncertainty Ellipse: a = semi-major axis in nmi.
 b = semi-minor axis in nmi.
 θ = orientation angle in degrees

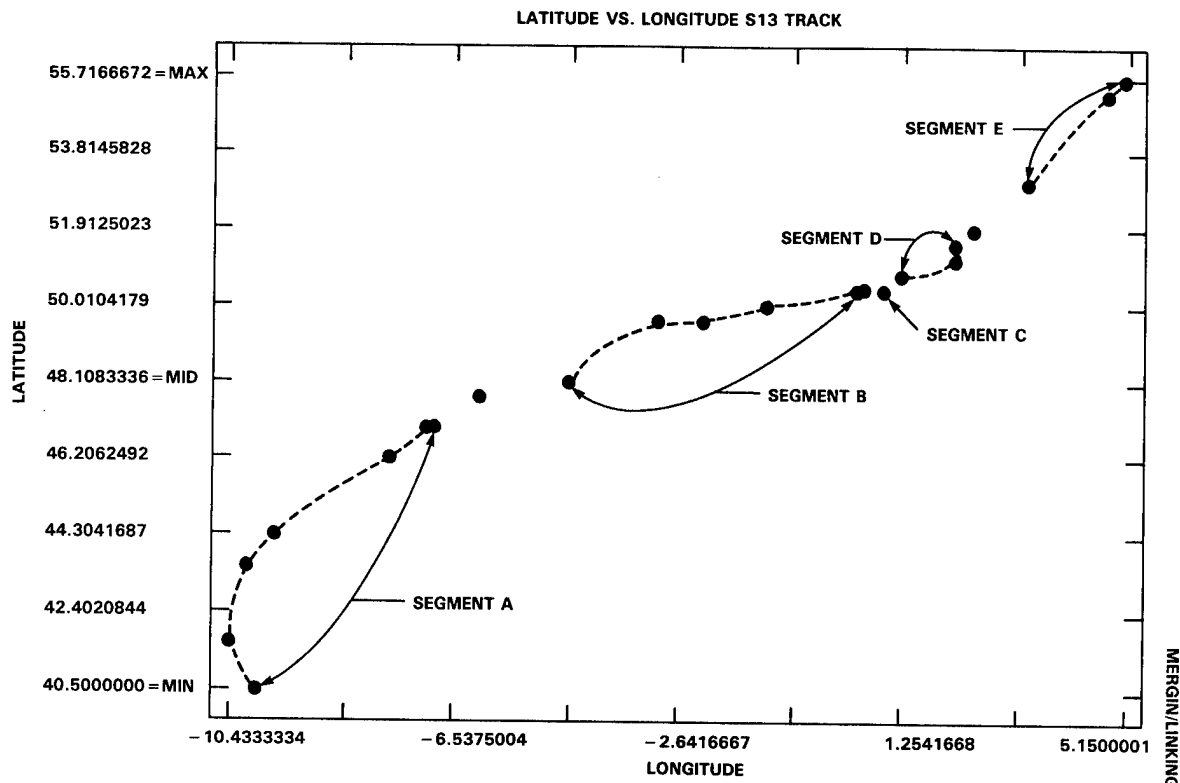


Figure 5

Segmentation of Track S13

TABLE II
Locations for Simulated Target I13

Time D/H/M	Location		Uncertainty		
	Lat	Lon	a	b	θ
0/20/21	4155	-1032	18	9	167
1/04/39	4353	-0959	13	10	88
1/05/50	4348	-0948	21	6	1
1/16/39	4620	-0745	14	2	69
1/18/53	4644	-0705	16	9	11
1/20/20	4702	-0707	8	5	7
1/23/51	4807	-0619	25	6	2
2/05/15	4853	-0458	8	7	111
2/11/54	4944	-0258	5	3	79
2/15/30	4954	-0204	23	13	102
2/20/20	5008	-0105	8	4	1
3/03/15	5024	00039	13	13	0
3/05/18	5024	00024	17	14	83
3/06/51	5026	00103	4	0	2
3/08/30	5037	00113	10	1	7
3/16/36	5113	00200	1	0	156
3/18/18	5123	00212	7	5	1
3/21/51	5152	00239	18	5	68
4/05/17	5309	00313	23	17	11
4/17/42	5459	00443	13	3	9

Time: D/H/M = Days/Hours/Minutes lapsed from zero time
 Location: Latitude, Longitude in Degrees and Minutes
 Uncertainty Ellipse: a = semi-major axis in nmi.
 b = semi-minor axis in nmi.
 θ = orientation angle in degrees

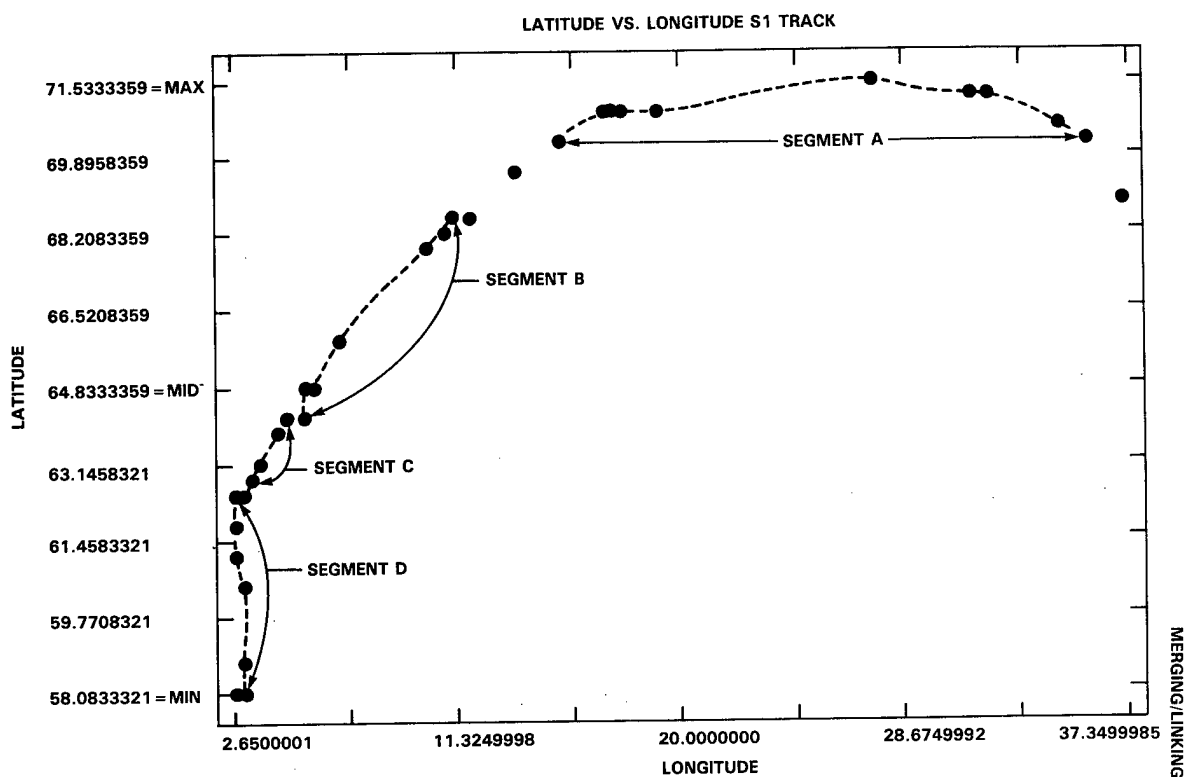


Figure 6

Segmentation of Track S1

TABLE III
Locations for Simulated Target P13

Time D/H/M	Location		Uncertainty		
	Lat	Lon	a	b	θ
0/19/37	4214	-1010	8	8	15
1/04/24	4420	-1001	7	4	2
1/06/47	4449	-0941	12	8	54
1/17/28	4656	-0736	16	10	3
1/18/41	4716	-0717	6	1	8
1/20/56	4745	-0634	12	10	39
2/00/05	4818	-0624	11	8	17
2/07/18	4949	-0359	7	4	63
2/12/07	5017	-0300	16	3	23
2/14/33	5020	-0231	16	9	1
2/21/09	5034	-0055	16	15	142
3/03/48	5053	00041	3	2	13
3/03/55	5052	00045	5	5	12
3/06/24	5058	00046	13	9	32
3/11/20	5111	00133	13	8	34
3/17/10	5144	00201	12	5	121
3/18/10	5200	00229	6	4	20
3/21/38	5228	00232	9	2	5
4/06/21	5353	00330	6	1	35
4/18/57	5548	00448	2	1	117

Time: D/H/M = Days/Hours/Minutes lapsed from zero time
 Location: Latitude, Longitude in Degrees and Minutes
 Uncertainty Ellipse: a = semi-major axis in nmi.
 b = semi-minor axis in nmi.
 θ = orientation angle in degrees

TABLE VI
Measure of Correlation for Segment Linking

Tester	Candidate	Computed Measure	Interpretation
S13A	S13B	0.689	Link uncertain
S13A	S13C	0.843	Accept link
S13A	S13D	0.835	Accept link
S13A	S13E	0.841	Accept link
S13ACDE	S13B	0.706	Accept link
S1A	S1B	0.890	Accept link
S1A	S1C	0.902	Accept link
S1A	S1D	0.756	Accept link
S13ABCDE	S1D	0.484	Reject link

TABLE IV
Measure of Correlation for Track Merging

Tester	Candidate	Computed Measure	Interpretation
S13	P13	0.495	Reject Merge
S13	I13	0.753	Accept Merge
I13	P13	0.500	Reject Merge

TABLE V
Locations for Simulated Target S1

	Time D/H/M	Location		Uncertainty		
		Lat	Lon	a	b	θ
S E G M E N T S	0/00/00	6852	03721	8	3	1
	0/12/11	7018	03606	11	5	37
	0/16/47	7030	03503	15	4	20
	0/17/45	7035	03457	2	0	44
	1/02/55	7109	03159	5	2	39
	1/04/06	7113	03123	1	0	42
	1/12/22	7135	02745	1	0	24
	2/07/03	7100	01919	4	2	44
	2/09/53	7056	01802	2	1	50
	2/10/41	7052	01739	3	1	18
	2/11/41	7050	01715	1	0	16
	2/15/23	7013	01532	1	1	57
	2/19/13	6930	01351	6	2	24
	2/23/43	6837	01157	21	8	46
	3/00/45	6824	01125	1	0	21
B	3/01/27	6816	01107	5	4	49
	3/03/20	6754	01018	12	5	39
	3/13/38	6546	00654	2	1	40
	3/17/32	6454	00543	4	4	46
	3/18/16	6446	00533	4	2	68
C	3/20/04	6409	00528	23	22	42
	3/21/14	6407	00445	11	6	34
	3/22/55	6343	00415	1	0	48
	4/00/58	6316	00339	5	2	0
D	4/02/33	6255	00312	2	2	60
	4/03/43	6238	00255	5	4	18
	4/04/31	6228	00247	4	2	48
	4/07/23	6145	00239	7	5	22
	4/09/15	6115	00244	20	7	38
	4/12/59	6018	00251	2	1	55
	4/19/21	5845	00255	6	3	70
	4/21/21	5815	00244	10	9	63
	4/22/10	5805	00252	3	1	74

Time: D/H/M = Days/Hours/Minutes lapsed from zero time
 Location: Latitude, Longitude in Degrees and Minutes
 Uncertainty Ellipse: a = semi-major axis in nmi.
 b = semi-minor axis in nmi.
 θ = orientation angle in degrees

5. EXAMPLES

The use of the Measure of Correlation is illustrated with two kinds of cases:

a. Merge Test. A test for deciding whether or not two or more tracks occurring in approximately the same time interval are associated with the same target.

b. Link Test. A test for deciding whether or not time separated tracks (track segments) are associated with the same target.

In both tests one of the tracks (or segments) is taken to be the Tester, each remaining track (or track segment), a Candidate. Best estimate values were obtained by the weighted, least squares cubic spline developed by The International Mathematics and Statistics Library. [6]

For each position an estimated one-sigma error in latitude and one-sigma error in longitude, computed from the uncertainty ellipse parameters, was used as a measured position weight in the best estimate latitude and best estimate longitude stages, respectively. For the smoothing value required by the program, half the quantity of input points was used per stage.

Merge Test

For the merge test three closely spaced tracks were simulated. (See Figure 4.) The three tracks came from "error-free" positions of two distinct targets, observed over the same time interval, and traveling parallel courses, one 36 nautical miles North of the other, Track P13 was simulated by adding random error and an uncertainty ellipse to sampled positions of the northern track. Tracks S13 and I13 were similarly formed from the other track. To simulate independent measurement systems, sampled positions of one track were chosen to be distinct in time from those of each of the other tracks. Simulated measurements are listed in Tables I, II, and III for S13, I13, and P13, resp. Table IV shows the measure of correlation obtained with each of the other tracks when track S13 is taken as the Tester. The computed measures indicate that S13 and I13 are tracks of the same target, and that P13 is a track of a target different from S13 and I13.

Link Test

For the link test two tracks were each divided into disjoint segments. Table I and Figure 5 display the segmentation of simulated Track S13. Table V and Figure 6 display the segmentation of simulated Track S1.

Segment A of Track S13 was taken as the Tester, segments B, C, D, and E, as Candidates. The link test results, given in Table VI, indicate segments A, C, D, and E belong to the same track. There is uncertainty about Candidate B being linked with Tester A. But when segments A, C, D, and E are linked to form the Tester, with segment B taken as the Candidate, Candidate B is accepted for linking.

Segment A of Track S1 was taken as the Tester, segments B, C, and D, as the Candidates. The link test results, given in Table VI, indicate that all segments belong to the same track.

Finally, segment D of Track S1 was taken as a candidate for the track composed of the linked segments A, B, C, D, and E of Track S13. As shown in Table VI, the link test rejected the candidate.

6. SUMMARY

An algorithm was presented for automating decisions on merging and linking of target tracks as represented in surveillance systems. The algorithm supplements established tracking techniques, providing the means for correcting two errors common to the techniques.

a. Error in merging tracks occurring in the same time interval.

b. Error in linking track segments.

Removal of these errors facilitates the correlation of one track with another. Application of the algorithm was illustrated with target location information representative of that found in established tracking systems. The examples showed that the algorithm gives correct decisions on the correlation of track to track, and that the algorithm is usable in the correlation of report to track.

7. AREAS FOR FURTHER STUDY

Acceptance/Rejection. As noted above, the acceptance/rejection thresholds were set on the basis of theoretical considerations, and are to be taken as subject to modification by results of a field study.

Tester Size. The algorithm is applied pairwise to sets of locations representing tracks. Because the method supplements established tracking techniques, at least one set of locations attributed to a target will contain two or more elements. A set that contains a single element cannot be chosen as the Tester.

It was found that the ambiguous "uncertain" decision could frequently be avoided when the larger set was made the Tester. In the case of testing for track merge, no limitation on the Tester size relative to that of the Candidate was observed, because in the simulations the sizes of the sets under comparison were of the same order of magnitude. But in the case of testing for the linkage of a single location with a set of locations covering a long time period (many days), it was found advisable to restrict the size of the Tester set so as to cover a shortened period of time when the single location was geographically close to the Tester track. A similar restriction may become advisable in testing two closely spaced sets for a merge. That restriction depends on the resolution of the method.

Resolution. After the establishment of empirically determined acceptance/rejection threshold values, it would be useful to determine empirically how far apart a pair of parallel tracks must be for the method to yield a correct answer in the merge test.

Detection of Incorrect Link/Merge. Testing of the algorithm covered many more cases than reported in this paper, but was not exhaustive, and uncovered no erroneous deci-

sions. However, one false decision to link or merge sets of locations could result in a sequence of false decisions involving those sets. But, the initial false decision introduces systematic error into the set representing the Mergence. And, within the resolution of the method, systematic error is detectable in the residuals that result from a maximum likelihood solution [8], the latitude-longitude residual set with greatest energy value being associated with the data set element responsible for the systematic error.

Though the weighted least squares spline may not be a maximum likelihood solution because the errors in the observations are not always Gaussian, the testing thus far indicates that detection of such a systematic error is feasible from residuals of the Mergence. It is expected that a study of field test data will make possible an acceptance verification test.

REFERENCES

1. Wiener, H. I., Willman, W. W., Kullback, J. H., Goodman, I. R., "Naval Ocean-Surveillance Correlation Handbook", 1978, Naval Research Laboratory 8340 (1979).
2. Goodman, I. R., Wiener, H. I., Willman, W. W., "Naval Ocean-Surveillance Correlation Handbook", 1979, Naval Research Laboratory 8402 (1980).
3. Reid, D. B., "An Algorithm for Tracking Multiple Targets", IEEE Trans. Automatic Control, AC-24, 843-854 (1979).
4. Mitzel, G. E., Barnett, P. G., Kuehne, B. E., Sommerer, S., "Wide-Area Correlation and Tracking of Surface Ships Using Multiple Sensors", Johns Hopkins APL Technical Digest, vol. 5, No. 1, 1984.
5. Reinsch, C. H., "Smoothing by Spline Functions", Numerische Mathematik 10, 177-183 (1967).
6. The International Mathematics and Statistics Library, Program ICSSCU, Cubic Spline Data Smoother.
7. Craven, P. and Wahba, G., "Smoothing Noisy Data with Spline Functions", Numerische Mathematik 31, 377-403 (1979).
8. Campbell, J. B., "Correction of Systematic Error in Tracking Systems by Means of Trend Analysis", IEEE 1978 Position Location And Navigation Symposium, 372-376 (1978).

AUTOMATIC CORRELATION OF MULTIPLE INTELLIGENCE SOURCES

Fred Schroeder and Larry Wright

BTG, Inc.
1945 Old Gallows Road
Vienna, Virginia 22180

Automatic correlation of intelligence information involves the evaluation and comparison of independent activity reports to determine if they are related to the same item of interest. The information that is used in this process is normally extracted from well-defined fields within formatted messages. In addition, correlators generally process only one type of intelligence information such as ELINT reports. This is based on the common characteristics of similar fields and the assumption that data contained within reports from different types of intelligence sources is difficult to compare. Comparison of information from several types of intelligence sources, however, can be supported by a number of related attributes. Therefore, it is possible to define an automatic correlator that not only utilizes these common attributes to relate the different sources but one which fuses the data to result in more information than is provided by any single source.

The concept for the automatic correlation of intelligence reports generated by multiple sources includes ELINT and imagery reports, each of which have strengths and limitations which complement each other. The ELINT reports are based primarily on the emanations of radar systems which are a major element of many military threats. These sources collect a large volume of data from emitters at world wide locations including both land and sea and report the information in the form of timely, well defined, fixed format messages. They are often limited in terms of location accuracy and specific functional identification of the site generating the emission. The emitter must be active and detected to be reported. Imagery reports are based on interpretation of imagery collected over a target area. These sources are not dependent on the activity of the site, as are ELINT systems, and provide more precise information including location and site functionality as well as information that is not provided by ELINT such as site elevation and operational status. Imagery results are also reported in the form of well-defined fixed format messages. Their two major limitations are the relatively small geographic area covered by any single image and the amount of time required to interpret the imagery and generate the reports.

These sources share a common set of attributes that serve as the basis for the fusion process. They include the time of the reported activity, the geographic location information such as latitude, longitude and BE numbers, as well as site characteristics such as ELMOT and weapon type within the ELINT reports and the equipment name within the imagery reports. By relating these sources based on the correlation of their common attributes and by fusing or combining the information, it is possible to create an automated system that uses the strengths of each source while minimizing their weaknesses. ELINT sources provide world wide coverage and support the near real-time identification of new or previously unreported sites and the tracking of mobile systems. Imagery sources provide more complete information as well as the site status. An automated multiple source correlation and fusion process is directly applicable to any military operation requiring near real-time information on identification, location, Order of Battle, and operational status of military threats.

Automatic Correlation Of Multiple
Intelligence Sources

Fred Schroeder and Larry Wright

BTG, Inc.
1945 Old Gallows Road
Vienna, Virginia 22180

INTRODUCTION

Effective mission planning and situation assessment require the use of time critical information to characterize enemy disposition and defenses must be considered during the planning process. BTG Inc. is actively involved in the development and deployment of computer systems that receive and correlate near real-time intelligence information and provide the resulting data to mission planning and situation assessment systems. Currently, we are involved in the specification and design of a data fusion system referred to as the Near Real-Time Threat Processor (NRTTP) to be included as an upgrade to a mission planning center (MPC). The MPC is responsible for route planning and mission preparation for a large number targets each year and is being upgraded to utilize many automated techniques. As part of the automated route planning process, the system will take into consideration the location and type of threats maintained by the NRTTP. This paper discusses the characteristics of the threats, the types and sources for intelligence information used to monitor their activities, the method used to correlate the individual sources and the process of fusing the multiple sources to form a more complete representation of the threat.

THREAT CHARACTERISTICS

Threats to the MPS missions include both afloat and land based defensive systems. Many of these, including those on land, are mobile and can change locations within short periods of time (hours). These threats can be divided into three major categories based on their degree of mobility. They are:

- a. Fixed sites
- b. Mobile with pre-planned sites
- c. Mobile with no pre-planned sites

The fixed threat sites include some types of surface to air missile (SAM) facilities and air fields. Since these sites

don't move, they can be studied over time by intelligence organizations to determine their location, functionality and operational characteristics. This information can then be provided to the planning centers and updated periodically. However, even though a historical view of the site can be maintained, it is necessary to use current intelligence data to determine the site's current status, particularly with regard to battle damage assessment.

The mobile threats with pre-planned sites include systems, such as some types of SAMs, that can be moved from one location to another but which require some type of physical site preparation. In general, these types of threats have many more prepared sites than weapon systems to occupy them. These pre-planned sites are characterized over time by intelligence organizations but the current locations of the weapon systems can only be determined by the analysis of current intelligence data.

The third type is the mobile with no pre-planned sites. These include ships and some types of land based systems including ZSU-23 AAA Guns and tactical SAMs. Intelligence organizations attempt to characterize the general operational areas for these types of systems but the current locations can only be determined by the analysis of near-real-time intelligence data.

Information related to all three of these types of threats is provided to the mission planning organizations in the form of Order of Battle Data Bases.

ELINT CHARACTERISTICS

Because of the mobility of the threats and the need for current status information, it is necessary for the planning center to augment the historical threat information with more current data collected and distributed by various intelligence sources.

A primary source of this type of information is electronic intelligence (ELINT).

ELINT systems are based primarily on radars which are a major characteristic of most defensive systems. Typical applications include target acquisition, tracking and guidance for the weapon systems themselves and as navigation for ships. Many of the radars are unique to a particular function or even a weapon system. In addition, some of the weapon systems not characterized by a single unique radar are identifiable based on a specific set of radars. ELINT systems are capable of receiving and distributing large volumes of data collected world wide. They are often very responsive and can distribute radar contact information very quickly. Because of this, ELINT collection systems are excellent sources for the detection, functional identification and location of threats.

BASIS FOR NRTTP ELINT CORRELATOR

ELINT information is used in a number of systems that are currently deployed to support planning activities. The two that form the basis for the NRTTP correlator are the Prototype Ocean Surveillance Terminal (POST) and the Prototype Analysis Workstation (PAWS). These systems are based on a common set of heuristics implemented within an expert system correlator that processes many forms of ELINT data and maintains or tracks the positions of ship and land emitters.

In order to discuss the correlation process, it is first necessary to define and briefly describe the characteristics or attributes reported by the ELINT sources. In general, ELINT information is reported as parameter values or distinct characteristics of emitters that serve to identify the individual emitter and its function. There are two basic types of ELINT parameters, descriptive and measurable. They are reported in textual messages that contain fixed format fields that support automated processing. An example of the Tactical ELINT (TACELINT) message format is shown in Figure 1. The top half of this figure contains the format for the message with the names of the

fields in parentheses, "()". Those that are used in the correlation process are underlined. The other values are as they would appear in an actual message. The bottom half contains the list of fields that are used in the correlator and examples of their content.

The descriptive parameters include general characteristics of the signal that was received, possibly resulting from other types of analysis. They include:

- a. ELINT Notation (ELNOT)
- b. Location
- c. Ellipse size and orientation
- d. Ship Name
- e. NOIC ID

The ELNOT is the type or class of radar assumed to have generated the related signal. The location and ellipse size and orientation provide geographic information. The ship name is often associated with afloat based emitters and is the platform from which the signal is thought to have originated. These parameters, especially the ELNOT and geographic information, are used within the correlator.

The correlator is primarily based on the measurable parameters associated with the ELINT information. These include:

- a. Radio Frequency (RF)
- b. Pulse Duration (PD) or Pulse Width (PW)
- c. Beamwidth (BW)
- d. Pulse Repetition Interval (PRI) or Pulse Repetition Frequency (PRF)
- e. Scan Rate (SR) or Interval

The RF is the transmitted frequency of the radar and can be used to support the determination of its use and physical size.

The PD or PW is the time that the radar is radiating energy and can be used to determine the minimum range of the radar and its range resolution (the ability to identify multiple targets close together).

EXAMPLE TACTICAL ELINT (TACELINT) REPORT FORMAT

```
UNCLAS
MSGID/TACELINT/(ORGNTN)/(MSG SER NO)/(MONTH)/(QUALIFIER)/(QUAL SER NO)//
COLLINFO/(COLL DIGRPH)/(COLL TRIGRPH)/(COLL MSN NO)/(COLL PRJ NAME)//
SOI/(TGT SGNL ID)/(DTCT TIM)/(TIM LOST)/(ELNOT)/(EMTR DSG)/(CTRY OF EVT)/(TRG ID)/(UNIT)/(WEAP)/(EMIT FNC)//
EMLOC/(ENTRY)/(LOC DATA CAT)/(LOC)/(RADIUS)/(ORIENTATION)/(SEMI-MAJ)/(SEMI-MIN)//
PRM/(ENTRY)/(BFW)/(RF OPR MODE)/(PRI-PRE)/(PRI ACTVY CODE)/(PD)/(SCAN TYPE)/(SCAN RATE)/(POLRZTN)//
```

Fields Used and Example Values

DTCT TIM	-	060812Z	ORIENTATION	-	027.5T
ELNOT	-	A101A	SEMI-MAJ	-	02.4NM
TRG ID	-	P00418002	SEMI-MIN	-	01.8NM
WEAP	-	SA-1	RF	-	827MHZ
LOC DATA CAT	-	F	PRI-PRF	-	PRI:002015.3
LOC	-	LS:512242N0115030E	PD	-	PD:3.200
RADIUS	-	03.7NM	SCAN RATE	-	4.2SPC

Figure 1.

The BW is a measure of the area coverage for the radar and is usually measured horizontally and vertically in degrees for the main beam or lobe. This effects the probability of the radar detecting an object and the accuracy of the elevation and azimuth measurements.

The PRI and PRF are inverses of one-another and measure the amount of time between radar pulses. This parameter is the bases for computing the Maximum Unambiguous Range which also is related to its operational use. Due to its consistency and unique values for a given radar, especially for those that are crystal controlled, it is one of the best correlation parameters.

The last parameter is the SR or Interval which is how often a target is illuminated by the beam of energy. This is dependent on type of radar and its function and is relevant for most types of radars. Again, due to its consistency and unique values for a given radar, it can be a very good correlation parameter and in some cases better than the PRI or PRF.

ELINT CORRELATION

For the purposes of correlation, the fields contained in the ELINT reports have been divided into three types. The first type is geographic, which consists of the position information such as latitude and longitude, and ellipse size and orientation. The second type is the non-parametric attribute, which consists of description information such as the ELNOT, Ship Name, and BEN.

The third type is the ELINT parametric information which consists of the PRI, PRF, RF, SR, PD, and PW.

Based on these three types of data (geographic, non-parametric attribute, and ELINT parametric) the correlator performs three types of processing. At each step the candidate contact reports are compared to the current contact being correlated and a score computed. The score represents a quantitative measure of how well the contacts match. The following is an overview of the type of processing preformed for each step in the correlation process.

The first step is geographic correlation which evaluates the previously reported contacts to identify a set of candidates that are geographically feasible. This is based on the geographic proximity, size and orientation of the error ellipses being compared, required speed to reach the new location, and for ships, the required course change. This results in a geographic score for each candidate.

The second step in the correlation process is based on the non-parametric attributes and is referred to as associative correlation. This process involves the comparison of the attribute values to determine if they are related to the same emitter. It is based on the number of attributes provided in the current and candidate report, and the number of matches. This results in an attribute score for each candidate.

The third and primary step in the correlation process is based on the ELINT parameters. This is also a multi-step process that compares the ELINT parameters provided in the contact report with those of previous reports. The first step is to convert the PRI or PRF to a base-band value. This is particularly useful for radars that vary the pulse interval or "count down" value based on a crystal oscillation rate. The base-band value then becomes a single value to be used in the comparison process. Some of these count down values can be found in Electronic Parameter Lists (EPLs). The correlator utilizes an emitter data base that contains the base-band PRI range for each class of emitter processed by the correlator. (Note: If an ELINT contact refers to an emitter that is not included in the emitter data base it is still correlated and tracked however, its description is not augmented by the additional data contained in the data base and its functional and operational characteristics must be approximated.) In cases where a given emitter class has slightly different measurable characteristics, the class is divided into versions based on the PRI ranges.

Once the base-band PRI is determined, an ELINT score is computed based on how well the PRI, SR, PW and RF of the current contact match the candidate values. The scoring includes the use of a parameter weight obtained from the emitter data base for each of the four ELINT parameters. (Note: If the candidate is not in the emitter data base, a set of default values is used.)

An overall total score is then generated by merging the geographic, attribute and ELINT scores and selecting a best correlation candidate. If only one candidate is found, the current contact report is merged to that track or site. If no candidates are found, a new track or site is created based on the current report. If more than one candidate is found, a complex rule based three (3) phase (geographic, non-parameter attribute and parameters) multi-pass process is performed which evaluates the individual scores singly and in combination with all three factors, a process from which the single best candidate is identified.

TEMPLATE MATCHING

When a new track or site is created for land based emitters, a special type of processing is performed to attempt to determine if the emitter is one of several emitters associated with a single operational element. Examples of this type of defensive system include SAMs that utilize several different types of radars such as those used for target acquisition, tracking and guidance. This process is referred to as template matching and utilizes a data base that describes the relationships of emitters used by different types of operational elements. The template matching process compares the new emitter and those in its geographic proximity with the templates identified in the data base. If there is no match, the emitter continues to be tracked as an individual. If there is a match, the emitter is grouped with the other related emitters and tracked as a group. This greatly increase the amount of information known about the defensive system by identifying the functional characteristics of the site (such

as a SAM site) as opposed to only knowing the use of the specific radar (such as range finding). It also increases the potential for proper tracking since an update to any of the individual emitter is also an update to the group and can be used to refine the sites actual position.

ELINT LIMITATIONS

ELINT sources are very responsive and provide a large amount of information, however, they are limited in several areas. A single ELINT report has limited location accuracy which is normally reported as an error ellipse that is measured in miles. This accuracy can be improved as multiple reports are combined and the overlap of the ellipses are evaluated. This process is shown graphically in Figure 2. This however, requires more time to collect the additional information. In addition, these reports do not contain certain types of data that are needed during the planning process. An example is elevation data. This is important to the precise positioning of the threat and therefore, the determination of its geographic operational constraints. For example, the threat may be located at the top of a hill or within a valley contained within the same error ellipse.

Probably the most limiting aspect of the ELINT sources is that they only report positive information (the contact of an emitters pulse) and can not report the absence of an emitter. This results in inactive emitters not being reported and emitters that have moved and not reported, to continue to be associated with their old location. (Note: For sea based emitters, an estimate of the current location can be made base on the historical course and speed.)

OTHER SOURCES OF INTELLIGENCE INFORMATION

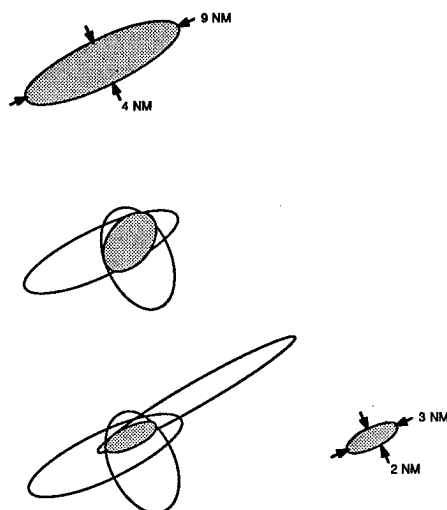
ELINT represents only one of many types of intelligence information available to the mission planning or situation assessment organizations. Others include Photographic Intelligence (PHOTINT), Imagery Intelligence (IMINT), Communication Intelligence (COMINT), Radar Intelligence (RADINT), Measurement and Signature Intelligence (MASINT), Visual Intelligence (VISINT), Telemetry Intelligence (TELINT), Acoustic Intelligence (ACINT), and Human Intelligence Sources (HUMINT).

These sources provide very different types of information in many formats and have very different response times. Of these, a major source of land based information is imagery. This information is reported in the form of a photograph or image and provided to human interpreters for evaluation. The imagery itself has little use in the automated correlation process. However, the result of interpreter's evaluation is normally a report that describes the content of the image. These reports include fixed format fields that are very suitable for automated processing.

IMAGERY REPORT CHARACTERISTICS

Imagery collections systems are able to provide more complete information related to the defensive systems. This often includes a more precise description of the type of threat, its location and addition physical and operational characteristics and is provided to mission planning organizations in the form of imagery interpretation reports. These reports do not contain the actual photographs but instead contain the results of the intelligence analysts evaluation of the imagery and often are available within

CONVOLVING ELLIPSES



- Each ellipse provides emitter location information.
- The convolving process combines the ellipse data based on past and present contacts to allow for a more accurate location estimation.

Figure 2.

hours of the receipt of the imagery. An example Imagery Interpretation Report (IIR) is shown in Figure 3. The top half of this figure contains the format for the message with the names of the fields in parenthesis, "()". Those that are used in the correlation process are underlined. The other values are as they would appear in an actual message. The bottom half contains the list of fields that are used in the correlator and examples of their content. These types of reports typically include the following types of data:

- a. Time On Target
- b. Geographic Location
- c. Elevation Of Target
- d. Target Name
- e. Object Count
- f. Target Category Code
- g. Order of Battle
- h. Order of Battle Equipment Name
- i. Basic Encyclopedia Number
- j. Operational Status
- k. Validity

The Time On Target and Geographic Location are basic information related to when the image was taken and the location of the object being reported. The Elevation Of Target is included for new sitings and represents a type of information that is difficult to obtain from other sources. The Target Name is determined based on the interpreters evaluation of the object and is related to the Basic Encyclopedia Number (BEN). The BEN identifies the specific site. The Target Category Code contains an entry from the Intelligence Data Handling System (IDHS) and is also related to the BEN. The Order of Battle (OB) contains an entry from a list of ten (10) possible types. The Order of Battle Equipment Name contains the standard OB type or NATO code. If there are multiple objects of the same type, the report normally includes an Object Count.

Two major components of the information provided by the imagery interpretation reports are the site status and the degree of certainty or validity associated with the reported information. The Operational Status provides both threat activity and inactivity information. It contains one of over twenty different status types which range from general to very specific. Examples include "operational" and "not operational" as general status, "occupied" or "unoccupied" in relation to pre-planned sites, and "dummy" in relation to radar sites used only for deception.

IMAGERY REPORT LIMITATIONS

Imaging systems also have a number of limitations. In general, they produce a smaller volume of data and require more time to process than ELINT sources and are normally less timely. In addition, they are subject to errors by the interpreter and include general types of information such as construction activity that are not related to a threat or threat status.

FUSING OF ELINT AND IMAGERY REPORTS

The combination of these two types of data, ELINT and imagery reports, can provide a very complete and accurate description of the type, location, and status of threats. The ELINT information can be used for both land and sea based defensive systems and provides world wide area coverage. It is the major source of information used to identify new threats not associated with fixed or pre-planned sites. This includes ships near shore and fully mobile SAMs. In addition, it can be used to identify active pre-planned sites and to track the movement of other mobile systems. The imagery report information is used to provide more accurate and complete information for land based threats. This includes location, elevation, and equipment type. In addition, it is used

EXAMPLE IMAGERY INTERPRETATION REPORT (IIR) FORMAT

```
UNCLAS
MSGID/IIR/(ORGNTR)/(MSG SER NO)/(MONTH)/(QUALIFIER)/(QUAL SER NO)//
RPTID/(RPT INDIC:EG. IPIR)/(RPTNG ORG)/(RPT SER NO)/(PRJ ID)/(IMG MSN NO)/(MSN DATE)//
HILITES/(MISSION HIGHLIGHTS)//
ITEMTYPE/(TYPE OF ITEM RPTD)//
ITEM/(ITEM NO)/(TARG NAME)/(BE NUMBER)/(TARG CAT)/(CTRY OF EVENT)/(NAT TSKNG INDIC)//
LOC/(GEO COORDS)/(UTM COORDS)//
OTID/(TGT LGTH)/(TGT WDT)/(TGT AZMTH)/(TARG ELEV)/(COMIREX REQ NO)/(MIL RGN CODE)//
STATACT/(OP STAT)/(SCNDRY STATUS)/(CHG SIGNIF)/(PRJ ID)/(IMG MSN NO)/(MSN DATE)/(AUTO TAC TARG GRPHC)//
RMK/(TGT-ACTVY REMARKS)//
OBID/(OB TYPE)/(DAT OF CNV)/(TIM ON TARG)/(ATTG-NBRG CMNT)/(CHG SIGNIF)/(PRJ ID)/(IMG MSN NO)/(MSN DATE)//
OBEO/(OB COUNT)/(VALIDITY)/(OB EQ NAME)/(OBJ CODE-FUTURE USE)/(OB EQUIP CMNTS)//
```

Fields Used and Example Values

TARG NAME	-	WESTERN ARMY BARRACKS	OB TYPE	-	MIS
BE NUMBER	-	BEN:0235-00235	DAT OF CNV	-	840602
TARG CAT	-	CAT:90000	TIM ON TARG	-	TOT:1207Z
GEOCOORDS	-	GEO:221600N1053012E	OB COUNT	-	0103
UTMCOORDS	-	UTM:18STN408898	VALIDITY	-	CNF
TARG ELEV	-	ELE:03580M	OBEO NAME	-	SA-6
OP STAT	-	OPR			

Figure 3.

to identify new sites not detected by the ELINT collection systems. Most importantly, it is used as a source of negative information such as a pre-planned site not currently occupied by its related weapon system or a fixed site not currently in operation due to maintenance activities.

This fusion process is based on the correlation of a common set of attributes. These include general types of information such as the time of the reported activity, the geographic location information such as latitude and longitude and BEN. It also includes specific site information such as the ELNOT and weapon type within the ELINT reports and the equipment name within the imagery reports. Figure 4 shows the relationship of the fields contained in the TACELINT and IIR reports, both to themselves and to each other. As shown, the left column shows the fields used to correlate ELINT reports. The first set of fields in the column is common to both TACELINT and SELOR RED (another ELINT message format). The second and third sets of fields are specific to each message format. The second column is the same as the first and the lines between them show which fields are related. The third and fourth columns are for IIR messages and the lines between them show their relationships. The final set of lines between the ELINT and IIR fields shows the interrelationships between the two sources.

The Target Id of the ELINT report contains either a Placename Identification Number (PIN) or a BEN. The PIN when combined with the geographic location, can also be translated to a BEN. This commonly defined value can then be used to relate the TACELINT Target Id to the BEN of the IIR.

The Order of Battle Equipment Name of the IIR reports the specific type of equipment seen in the image based on the IDHS equipment list. The Weapon Type of the ELINT report is selected from a weapons list. Based on the analysis of the list of possible field values contained in the related instruction manuals used to prepare these reports, it is possible to generate a translation table that converts the field values of one report to those of the other. This translation table then defines the relationship between the two reports.

The NRTTP refers to this translation table as the Attribute Data Base. This data base also contains supporting information that augments the content to the fields. An example of how this translation process is performed is shown in Figure 5. This example shows a subset of the fields from an IIR and TACELINT report. The Attribute Data Base is used to translate the SAM-6 equipment name of the IIR and the SA-6 weapon type of the TACELINT to the common correlator value of SA6. In addition, the values are augmented by their respective OB types. (Note: Due to similarities in values, in some cases it may be possible to develop a translation algorithm that does not require every possible field value to be pre-defined in the data base. Our design currently identifies a generic table which could be replaced or augmented by a process of this type at a later time.)

The Order of Battle Equipment Name of the IIR report is also related to the ELNOT of the TACELINT report. This occurs when the IIR references a radar system by name which can then be associated with its ELNOT by the

SOURCE DATA RELATIONSHIPS

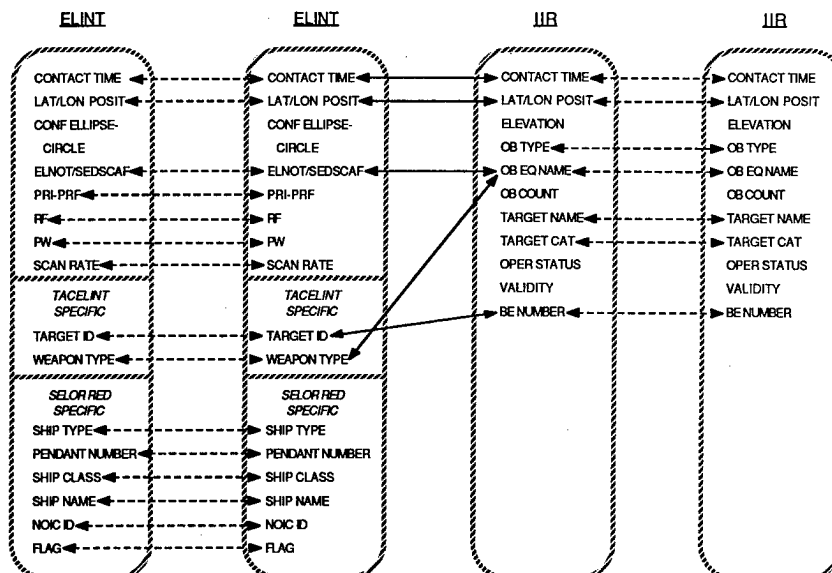


Figure 4.

correlator. This then allows the IIR to be indirectly related to the ELNOT of the TACELINT report.

Given the relationships described above the fusion process is very similar to that performed by the ELINT correlator. The fields of the two types of reports are divided into three categories. These are geographic, attribute and emitter. While the ELINT reports normally contain all three types of data, the IIR fields are primarily categorized as geographic and attribute. In special cases as describe above, it is possible for the IIR equipment name to be converted to an ELNOT which results in it being categorized as an ELINT field. Once the fields have been categorized, the geographic and attribute values are translated into a common set as described above. This includes converting the geographic ELINT PIN into a BEN and translating the attributes into a common nomenclature. The correlator then generates an attribute, ELINT and geographic score as done for ELINT correlation. These are then used to select candidate matches and finally to determine the disposition of the contact report being processed. Once the correlation has been performed, the related track or site entry is updated with the new information. In the case of an imagery report, the entry is also updated with its status.

SUMMARY

Automatic correlation and fusion of ELINT and imagery reports will provide the MPC with a very complete and timely location and operational status information related to the defensive systems that are threats. Due to its general characteristics, this capability is applicable to all mission planning activities that require time critical information. In addition, it is possible to include other types of intelligence data by identifying the relationships between the reported information and expanding the set of rules for fusing them, as was done for the TACELINT and IIR reports.

EXAMPLE USE OF ATTRIBUTE DATA BASE

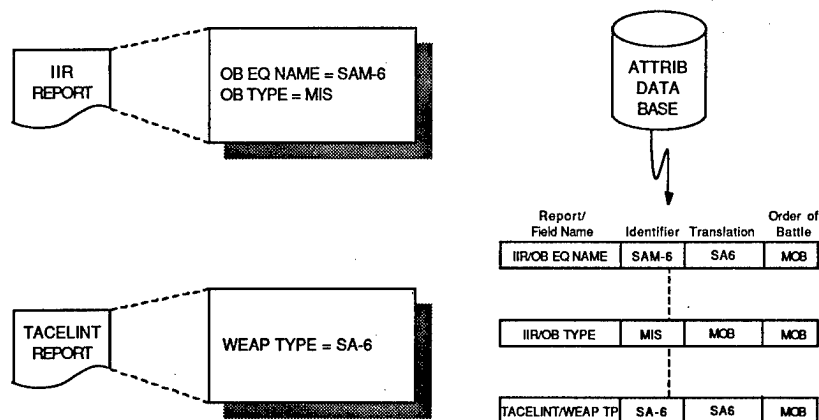


Figure 5.

An Adaptive Hull-to-Emitter Correlator

Carey Priebe
David Marchette

Command and Control Department
Code 421
Naval Ocean Systems Center
San Diego, Ca. 92152

INTRODUCTION

One of the functions of a data fusion system involves identifying the platforms reported in Electronic Intelligence (Elint) data. These reports contain a set of parameters measured from the platform's radar. The assignment of a name to this radar report is called Hull-to-Emitter Correlation (hultec).

In [4] the authors described an Adaptive Network System (ANS) (popularly referred to as "neural network") approach to this problem. The particular paradigm used was a backpropagation network. This system classifies the data into linearly separable classes, using a gradient descent, least mean squared error algorithm to determine the separating hyperplanes. The network returns a value for each hull that gives a measure of its distance from a point in the teaching set (actually it's a measure of the distance from the separating hyperplane). This approach compared favorably to traditional statistical techniques. One problem with this approach is that the output of the network, though it ranges continuously from zero to one, has no relationship to probability. Although the network gives a ranking for the hulls it has learned, the value associated with any one hull gives little information. It is only the value relative to the other hulls that can be used in further decision making.

This paper deals with a modification of a statistical technique that retains the distributed nature of the ANS, while allowing the network to learn to approximate the probability distribution of the data. This not only allows the network the potential of better performance, but it allows other systems to use the output of the network in a probabilistic sense.

THE HULTEC PROBLEM

For the purposes of this paper, a report is a set of parameters measured from a radar signal: PRI (pulse repetition interval), Scan (scan rate), RF (frequency), etc. The hultec problem is to determine the emitter of origin for the report, based on these measured parameters. By its nature, this is a difficult problem. Errors in measurement are bound to occur, so the system requires a measure of fault tolerance. Some sensors report different parameters than others, requiring the system to be able to handle missing data. The parameter spaces of several emitters may overlap, so the system must respond with a ranked list of emitters, rather than a single emitter.

The main problem with the backpropagation approach investigated in [4] is that it attempts to assign a single hull to the report. Often, as mentioned above, the distributions of the hulls are not disjoint. It is therefore not always possible or desirable to assign a report to a single hull. Instead, one would like to provide a list of the possible hulls with some measure of the likelihood of classification associated with each hull. One way to

do this is to model the probability density function associated with each hull. In other words, the reports define a sample space for each of the classes (hulls), and the reports associated with each class define a probability distribution for that class over the sample space.

An improvement over the ANS approach described in [4] is one which can learn the probability density functions for the classes. This is the approach taken by the network described below. The output from the net is the value of the probability density functions for all the hulls evaluated on the report. The value output for a given hull, in contrast to that given by the backpropagation scheme, has a meaning in an absolute sense as well as relative to the values output by other hulls. The network can indicate how certain it is that a given input stimulus belongs to a given class, thereby indicating, for example, that the second place choice for classification is very high indeed and should not be disregarded. This gives more information than a winner-take-all approach, and this information could be used in conjunction with other information (such as a priori information, data from other sensors, etc.) to improve overall performance.

Another important advantage that a probability density estimator gives over a winner-take-all classifier is the ability to detect unknown emitters. When the output is low for all the emitters known by the system, it is likely (though not certain) that the report comes from an emitter that is unknown to the system. The ability to recognize new emitters can be as important as the correct classification of known emitters, and a system that can continue to cluster reports from this new emitter is of great value to a data fusion system.

KERNEL ESTIMATORS

The problem of probability density estimation has been attacked by many different methods. One of the more successful is the technique of kernel estimation [5]. Consider a one dimensional problem, with a single class. The goal is to approximate the probability density function for the teaching data, which consists of a collection of points drawn from an unknown distribution. In its simplest form, the kernel estimator involves assigning to each of the teaching points a gaussian distribution, the kernel, with mean equal to the input point and a fixed variance, called the window width. This produces, as a partitioning of the input space, a Voronoi diagram [1], the optimal nearest neighbor partitioning. These gaussians are summed, weighted by the window width and the number of teaching points. Therefore, the kernel estimator with kernel K is defined by

$$f(x) = \frac{1}{nh} \sum K \left(\frac{x - X_i}{h} \right) \quad (1)$$

where h is the window width. Distributions other than the normal may be used for the kernel K , but for the purposes of this paper, the kernel will always be a normal distribution.

The ANS architecture proposed, an implementation of the kernel estimator, assigns a node in the hidden layer for each gaussian (see Figure 1). The node is connected to the input layer, with the weights set to the mean of the gaussian. These connections do a subtraction, rather than the traditional multiplication of their signal, and this signal is then summed by the node and the kernel is applied. In the case of a single dimensional input the node retains a variance, while in the multidimensional case, the node may retain a covariance matrix, or it may keep a single variance.

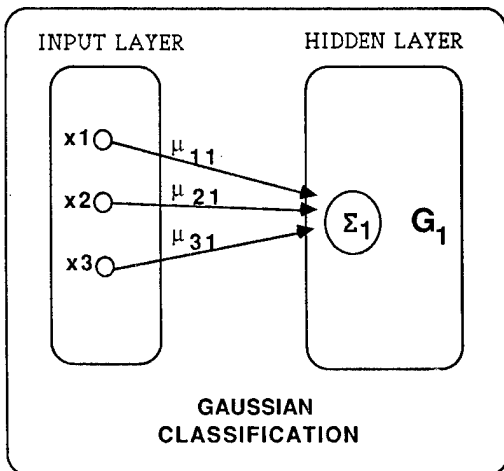


FIGURE 1

Representation of Gaussians

The gaussians are presented with the n -dimensional input from the input layer having n nodes and independently compute their respective activation values (gaussian distances) from this input. The activation value for these gaussian nodes in the hidden layer can be obtained via the following formula:

$$G_j(x) = \frac{\exp(-0.5[(x - \mu)^t \cdot \Sigma^{-1} \cdot (x - \mu)])}{(2\pi)^{(d/2)} * |\Sigma|^{(1/2)}} \quad (2)$$

Here $G_j(x)$ is the activation value of the j th gaussian node G_j when presented with vector input x . Σ is the covariance matrix for gaussian node G_j , while μ is the vector-valued mean for this gaussian. d is the dimensionality of this particular gaussian and is equated with the dimensionality of the input layer, i.e. the number of parameters in the report. Since the components of the mean μ are represented as a node's input weights, these weights can be thought of as shifting the origin for the node. A gaussian function is then applied to the inputs of the node.

If the matrix Σ is diagonal, then (2) is just the product of d independent gaussians, one for each input. This observation gives a method for coping with missing data: "drop" the appropriate gaussian from the product. In (2) this corresponds to assigning the missing input the value at the mean for that component. The system will continue to work as if the data were there. This approach gives an overly optimistic output for the report, but the ranking will be reasonable.

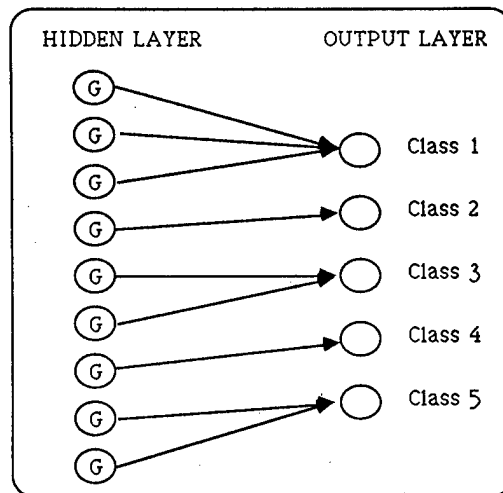


FIGURE 2

Combination of Gaussians

The outputs from the hidden layer are summed, weighted by the number of points in the node, to produce a final answer (see figure 2). There is one node on the output layer per class. If there is more than one class stored within the network, the nodes on the final layer are connected only to the nodes corresponding to their class. This method of combining gaussians to produce an overall distribution provides the system with enormous flexibility. Although the initial kernels being used are predetermined to be gaussians, many widely varying distributions can be approximated by a sum of gaussians (see Figure 3). The overall distribution for a particular class is not, therefore, assumed to be normal. In this manner the network can be used as a classifier, giving output that is meaningful in a probabilistic sense.

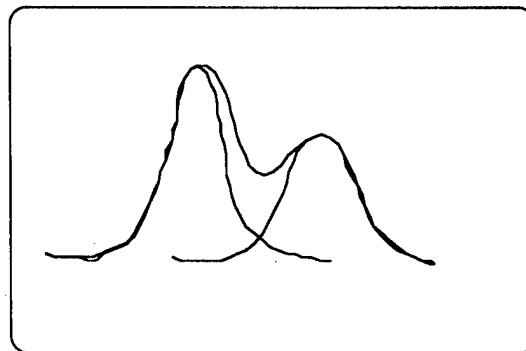


FIGURE 3

Sum of Gaussians

One difficulty with the kernel estimator is the choice of window width. A width that is too wide will produce a unimodal approximation, while a narrow choice can produce a multimodal distribution with inherent spikes. The width can be chosen experimentally, or one of the many adaptive kernel estimation algorithms may be employed (see [5]). A modification of the kernel estimator which allows the window to adapt to the data is described below.

MOVING MEAN AND VARIANCE

The kernel estimator described above is static once the initial learning is finished. One way to modify this scheme to produce a dynamic system is to allow the nodes to vary their means and covariance matrix as the data is received.

Only those nodes whose activation values reach some threshold defined by that node's activation function (usually the value of the gaussian at a fixed multiple of one standard deviation from the mean) are considered to be a likely category for the current input and are updated. This updating consists of moving the mean and variance of the gaussian based on the current input and some measure of the total number of inputs to the gaussian thus far. For the updating of the mean, we have

$$\mu(j+1) = \mu(j) + [1/(j+1)][I(j+1) - \mu(j)] \quad (3)$$

where $\mu(j)$ is the (one-dimensional) mean after the j^{th} input and $I(j+1)$ is the $j+1^{\text{st}}$ input to be categorized in this gaussian. The updating of the covariance is similar. Here

$$S_{xy}(j+1) = S_{xy}(j) + [j/(j+1)] * A_{xy}(j+1) \quad (4)$$

with

$$A_{xy}(j+1) = (x(j+1) - \mu_x(j)) * (y(j+1) - \mu_y(j)) \quad (5)$$

Then

$$\Sigma_{xy}(j+1) = \frac{1}{j} S_{xy}(j+1) \quad (6)$$

where we are calculating $\Sigma_{xy}(j+1)$, the x,y component of the covariance matrix Σ after the $j+1^{\text{st}}$ input clustered in this Gaussian. $x(j)$ and $y(j)$ are the x and y component of the input vector after the j^{th} input, and μ_x, μ_y are components of the mean. These formulas require the node to store j , the number of points in the node. This value provides the weight used in the sum of nodes computed for the final layer: j divided by the total number of points in the class. In this manner, the nodes corresponding to clusters near the mean of the distribution are weighted higher than nodes corresponding to outliers.

This approach can also be made to choose the number of nodes in the hidden layer. When a point comes in, the hidden nodes all evaluate their response to the input. If none of the nodes fire above a threshold, it is assumed that none of the existing nodes properly categorizes the input and a new node is created. Otherwise, all nodes of the appropriate class that fired above a threshold are allowed to adjust their parameters.

One of the advantages of this system is that it is dynamic. If the distribution of the data is drifting with time, the network's distribution will also drift.

The kernel method, and the moving parameters method, can be modified to allow the network to continue in an unsupervised manner. This allows the system to continue to learn, even though truth data may not be available. This is important in many Navy applications, as truth data is often hard to come by, and a system that clusters like entities may be the best that one can hope for.

The idea behind the unsupervised learning rule is that if the system has no a priori knowledge pertaining to the class of the input, it should update each class proportionally to the likelihood that the input comes from that class. This is indicated in equations 7-10. Equations 7 and 8 are the usual update rule for the mean (note that this is written in a less efficient but conceptually simpler form than equation 3):

$$\mu_{\text{new}} = \frac{j * \mu_{\text{old}} + x}{j + 1} \quad (7)$$

$$j = j + 1 \quad (8)$$

Here j is the number of points, and x is the input. In the case of an unknown input, the value of the distribution is used to weight the amount of change:

$$\mu_{\text{new}} = \frac{j * \mu_{\text{old}} + f(x) * x}{j + f(x)} \quad (9)$$

$$j = j + f(x) \quad (10)$$

A similar rule is used for the variance. The variable j is now a floating point number rather than an integer, but the two approaches are very similar in their implementations. This is essentially the Bayesian approach described in [7].

The unsupervised learning rule allows the system to continue to learn when truth data is unavailable, producing clusters of "like" reports. A typical use of this rule would be to first teach the system on known inputs using the supervised learning rule, then allow the system to monitor data, modifying its distributions using the unsupervised learning rule.

RESULTS

The kernel estimator was compared against CARI (Computer Assisted Radar Identification) [3], a classification program based on standard statistical techniques, on a data set consisting of 4019 points taken from 63 hulls. The parameters reported are PRI and Scan. The two systems were cross-validated using this data set: each system is taught on 4018 reports and tested on the missing report. If the hull with the largest value on the output corresponds to the hull from the missing report, the system is said to be correct. If the second highest output corresponds to the correct hull, the system gets a second place, and if the third highest is correct the system gets a third place. This is repeated until the system has been tested on all the reports in the data set.

The results are tabulated below. Note that the kernel estimator is slightly better on this data set. Preliminary indications are that the adaptive mean and covariance approach can in some cases improve on the kernel estimator, but in general it requires more points to construct its estimate.

Method	Correct	Second	Third
CARI	79%	16%	3%
Kernel	82%	12%	3%

These implementations were run on a PC AT compatible computer. It should be noted that the backpropagation algorithm described in [4] could not be cross-validated on this data set, even using a Cray, due to the prohibitive processing requirements of the learning algorithm. Although this is irrelevant for the finished system (after teaching), it is an important consideration for a system intended to be dynamic, and hence continually learning.

CONCLUSIONS

The classification scheme described herein develops a statistical model of the input entering the system. In addition, any a priori knowledge about the distribution of the incoming data can be incorporated into the system. This yields a powerful, dynamic classification tool possessing the advantages inherent in network methodologies and yet is readily understood via statistical analysis. The system can operate in both a supervised and unsupervised mode, and can handle missing as well as noisy data.

The gaussian nodes which are integral to the system are independent of one another. This allows each node to independently perform its processing based on its inputs and its

activation function. In keeping with traditional neural network schemes, this yields an easily parallelizable system. Each node can be thought of as an individual, independent, primitive processor. The nodes in various layers of the net are then linked through weighted connections to nodes in the previous layer. Thus the processing time in a hardware implementation is independent of the number of nodes required to estimate the density, and it is independent of the number of hulls in the system.

Another aspect of the system's dynamic nature is the adaptive network size. Within hardware constraints, gaussian nodes can be allocated dynamically, as dictated by the statistics of the incoming data. Under certain learning conditions, many gaussian nodes may be necessary to capture the salient features of the data. At other times, the actual distributions being modeled may resemble unimodal gaussian functions, thereby requiring very few gaussian nodes.

It may be useful to consider a more mature system in which information entering the system at different times from multiple reports can be used to strengthen the classification process. A temporal classification system employing the gaussian scheme described herein has been developed [6]. While one report may not give enough information for a classification, a track of several reports may give more information. Another application of this approach would be to look at the signal itself rather than measured parameters of the signal, and classify it based on the temporal nature of the signal.

REFERENCES

1. Aggerwal, Chazelle & Guibas, "Parallel Computational Geometry", (1985), *Proc. of the 26th IEEE Symp. on the Foundations of Comp. Sci.*, 468-477.
2. Chan, Golub & LeVeque, "Algorithms for computing the sample variance: analysis and recommendations", (1983), *The American Statistician*, (37) 242-247.
3. Kramer, "A Survey of New Techniques for processing Electronic Intelligence", (1987), *1987 Tri-Service Data Fusion Symposium*, Vol. II, 1-16.
4. Marchette & Priebe, "An Application of Neural Nets to a Data Fusion Problem", (1987), *1987 Tri-Service Data Fusion Symposium*, Vol. I, 230-235.
5. Silverman, Density Estimation, Chapman and Hall, 1986.
6. Sung & Priebe, "Temporal Information Processing", (1988), Submitted to *1988 Int'l Conf. on Neural Networks*.
7. Titterton, "Updating a Diagnostic System using Unconfirmed Cases", (1976), *Appl. Statistics*, 25, 238-247.

THE KNOWLEDGE WORKBENCH: AN AI RAPID PROTOTYPE TESTBED

Randall W. Hill, Jr.

Jet Propulsion Laboratory
4800 Oak Grove Drive, M/S 126-200
Pasadena, CA 91009
(818) 354-6641

Current approaches to building large data fusion software systems involve a long and costly process: years may elapse between the initial system concept to the time that the system is put into the user's hands. By the time the system is fielded, the technology is dated and changes or improvements to the system may be years in coming. Another approach to understanding and building data fusion software is to rapidly build prototypes, obtain user reaction to the prototype, and iteratively improve the system. This paper describes a set of rapid prototyping tools we have developed called the Knowledge Workbench. The Knowledge Workbench is a flexible testbed for developing and experimenting with data fusion architectures; it includes an object-oriented computing environment, a knowledge-based expert system shell, and a blackboard-style architecture. Collectively, the testbed tools provide the capability to rapidly prototype a data fusion system by giving the user the ability to design and implement a system architecture using knowledge sources to capture the functional knowledge and message-passing to implement the system data flow. A knowledge source contains procedural or rule-based knowledge pertaining to a function within the system; it is encapsulated as an object in the computing environment, thereby maintaining the function's integrity by keeping the knowledge in a modular form. We illustrate the use of the Knowledge Workbench by describing a tactical intelligence analysis system prototype we built which processes blocks of TACREP messages, performs analysis on the contents using a set of knowledge sources, and generates the intelligence product as a set of JINTACCS messages.

THE KNOWLEDGE WORKBENCH: AN AI RAPID PROTOTYPE TESTBED

Randall W. Hill, Jr.

Jet Propulsion Laboratory
4800 Oak Grove Drive, M/S 126/200,
Pasadena, CA 91109

INTRODUCTION

This paper describes a set of rapid prototyping tools we have developed called the Knowledge Workbench. The Knowledge Workbench is a flexible testbed for developing and experimenting with data fusion architectures; it includes an object-oriented computing environment, a knowledge-based expert system shell, and a blackboard-style architecture. Collectively, the testbed tools provide the capability to rapidly prototype an intelligence data fusion system by giving the user the ability to flexibly define "knowledge components" and the data flow among them. A prototype's functional knowledge is encapsulated in a set of loosely coupled knowledge components, and the data flow among these components is implemented using the message passing facilities provided by the object-oriented environment.

The underlying goal of the Knowledge Workbench was to develop a flexible set of tools for use in prototype development. This goal was motivated by the fact that most intelligence data fusion systems take years to develop, and once developed they are often already severely dated both in terms of technology and technique. By providing a better set of tools for prototype development we hope to improve the ability to experiment with intelligence data fusion architectures, thereby improving the quality of the requirements definition for such a system.

We will first present the motivation for our approach to developing the Knowledge Workbench by characterizing the data fusion problem and how it is viewed from the perspective of Distributed Artificial Intelligence (DAI). Next, we will describe the Knowledge Workbench architecture by giving an account of the system's most salient features and how they contributed to the testbed. Finally, we will demonstrate the utility of the Knowledge Workbench by describing a prototype that was built using the tools in the testbed.

MOTIVATION

There were two underlying motivations for the design of the Knowledge Workbench: (1) develop a set of tools fitting the characteristics of the intelligence data fusion problem, and (2) create a flexible testbed environment where a variety of analytical and computational techniques can be used within the same prototype. In this section we first describe how we characterize the intelligence data fusion problem in order to explain the design choices we made. Then we describe some technology background which relates to the class of problems characterized by intelligence data fusion.

Intelligence Data Fusion Problem Characterization

Our view of the intelligence data fusion process derives itself from the functional and organizational perspective of Army tactical intelligence collection and analysis activities. In particular, we are concerned with systems which can be used to support tactical operations which must:

- (a) produce a view of the battlefield using single discipline intelligence analysis,
- (b) combine the views from all single discipline intelligence analysis activities into a coherent view of the battlefield, and
- (c) plan the use of sensing resources so as to maximize their effectiveness in collecting data to support the focus of the analysis.

Each of the analysis activities involves processing streams of data from a distributed set of sensors or sources which may vary in capability and type and which provide a temporally disparate view of the battlefield. Taking this into account, "classical" intelligence data fusion proceeds through a number of steps whereby the sensor data is aligned, associated, correlated and fused

into a composite object or view. There are a number of different approaches to the actual sequence of steps and techniques applied but it isn't our intent to discuss them here, rather, we wish to point out the characteristics of the intelligence data fusion problem domain which led to our design choices:

- (a) Large grain problem - Although this is a qualitative description of the domain, it implies that the fusion problem is large enough to be easily decomposed into a set of sub-problems, each requiring a high degree of special knowledge or resources. This implies that the sub-problems are decomposed along analytical lines so that a high level of communication with other sub-problems is not required. Some readily identifiable intelligence data fusion sub-problems are: correlation, entity identification, entity attribute analysis, entity relationship analysis, aggregation, and node analysis.
- (b) Loosely Coupled - Once decomposed, the sub-problems display a moderate to high degree of independence. There are some interdependencies in terms of the sequence of processing, but the sub-problems can, in some cases, be solved concurrently, thereby removing the need to synchronize the sub-problems into a lock-step sequence. Instead, the sub-problem solvers may be able to opportunistically update or query one another for information.
- (c) Data-driven - By nature, the fusion problem is data-driven due to the fact that there is a constant stream of new sensor data channeled into the fusion center. This implies that solutions are temporal and open-ended, and a solution may be triggered or changed by the presence of new information.
- (d) Variety of Processing Paradigms - Although this is not characteristic of the problem domain, rather, of the existing approaches to dealing with the domain, it is a major consideration for building a testbed. The current approaches to intelligence data fusion include algorithmic and numeric computation, fuzzy or probabilistic reasoning, template matching and knowledge-based methods, to name a few. No single approach will solve all of the sub-problems, so there is a need to construct hybrid systems which take advantage of the applicable techniques in each area.

Distributed Artificial Intelligence

The intelligence data fusion problem characterization fits well with some of the approaches to problem solving used in the

burgeoning field of Distributed Artificial Intelligence (DAI). Much of the research in Artificial Intelligence (AI) has focused on building a single intelligent agent which solves problems using heuristics, knowledge-based reasoning, natural language understanding, planning, or theorem-proving. The single agent paradigm can be effective in solving problems in a closed domain, but it does not fare as well in cases where there are multiple interlinking problems in a domain spanning several areas of expertise, or in this case, intelligence disciplines and analytical techniques. For example, it would be unthinkable to implement a single expert system to perform all of the intelligence data fusion functions within the problem domain we mentioned earlier.

With the advent of parallel computer architectures and advanced networking techniques, it has become possible to create distributed multi-agent problem solving systems. Multi-agent problem solving systems attempt to combine information from a diverse set of sources to solve problems; the work is divided among a number of problem solving components which cooperate at the level of dividing and sharing knowledge about the problem and the solution state (Ref. 1, 7).

Some of the early work in DAI resulted in a problem solving architecture known as a Blackboard System. A Blackboard Architecture represents partitions of expertise in modules called knowledge sources. The knowledge sources attempt to cooperatively solve a global problem by sharing intermediate problem solutions on a central data structure known as a blackboard. They communicate with one another indirectly in that all of their communication is done via the blackboard structure. A knowledge source monitors the blackboard with a set of pattern triggers; if there is a match between a knowledge source's pattern trigger and some data on the blackboard then a control mechanism is notified. After all of the knowledge sources have an opportunity to view the blackboard, the control mechanism chooses which knowledge source to activate based on some conflict resolution criteria. Blackboards have been used to implement a number of different systems including the Hearsay II project (Ref. 4) and the Distributed Vehicle Monitoring Testbed (Ref. 5).

In conclusion, the goals and objectives of DAI are very closely aligned with the characteristics of the intelligence data fusion problem domain. The DAI concept of a knowledge source can be used to implement the sub-problems which resulted from decomposing the large grain fusion problem. Since the fusion sub-problems tend to be loosely coupled, the DAI concept of blackboard problem solving seems appropriate also. In fact, we have adopted some of the concepts from the blackboard approach to problem solving because of the leverage it provides in terms of partitioning the large grain data fusion problem into a set of what we call knowledge components and the concept it provides of cooperative problem solving. Two of the primary differences between a

classical blackboard architecture and our approach are (1) the way in which the knowledge components communicate, and (2) how they are activated. This is discussed in more detail later in the paper.

KNOWLEDGE WORKBENCH ARCHITECTURE

As previously stated, the underlying goal of the Knowledge Workbench is to provide a flexible testbed for constructing, testing, and modifying intelligence data fusion prototypes. In order to meet this goal we had to develop the testbed architecture so as to account for the characteristics of the data fusion problem domain, while at the same time exploiting current approaches to problem solving and computation as it has been developed in DAI and other disciplines.

Using the metaphor of a workbench, we see the testbed as a way of constructing an intelligence data fusion prototype in the same way as one might assemble a piece of furniture: first the parts are made and then they are fastened together. The added requirement for the testbed is that one should be able to disassemble the prototype and connect the pieces another way. In the description that follows we will explain how we model the fusion sub-problems as "knowledge components" in the architecture; the knowledge components are "glued together" by the data flow. The ability to treat the knowledge components as self-contained and modular is provided by the object-oriented environment; in addition, the data flow is implemented using the message passing facilities also provided by the object-oriented environment. The result is that the Knowledge Workbench provides a flexible testbed for prototype development. We define flexible as (1) the ability to insert, remove, or exchange knowledge components within a prototype architecture, and (2) the ability to easily change the data flow among knowledge components.

Object-Oriented Environment

One of the critical design choices we made for the Knowledge Workbench was to embed the testbed in an object-oriented environment. The reason this choice was critical is that the object-oriented programming paradigm underlies the entire system and provides it with many of the features required for the intelligence data fusion problem characterized above.

Object-oriented programming is a concept that was first realized in SIMULA, a simulation language, and has since been used in other languages such as Smalltalk, Flavors and Loops. Smalltalk goes to an extreme in terms of providing a totally object-oriented environment (Ref. 3), while Flavors and Loops are object-oriented features developed for use in the Lisp environment. Our object-oriented shell was developed to simulate the Time Warp distributed operating system (Ref. 6).

The basic component of an object-oriented system is the object. An object is an entity which combines the properties of data and procedure: it preserves local state and it performs computation (Ref. 8). In keeping with this definition, all of the problem solvers and data in the Knowledge Workbench are implemented as objects. All communication among objects is through the message passing facilities provided by the object-oriented shell. Hence, all activity in the Knowledge Workbench is initiated by passing a message to a knowledge component as opposed to invoking a procedure.

Memory is not shared among objects; each object has a private data space. In order to share data, an object must send the data in a message to another object. In this way, objects cannot directly access the data space of one another. The messages are time-stamped with the sending and receiving times, and they also contain the names of the sender and receiver. The time stamps are used to synchronize the activation of the objects; an object can only read a message when the global time equals the receiving time of the message.

Knowledge Components

Data fusion functions are conceptually treated as "knowledge components" on the Workbench. A data fusion "knowledge component" can be either procedural or declarative in nature, thereby allowing the prototype engineer to mix algorithmic and probabilistic techniques with knowledge-based approaches to problem solving within the same prototype. In a practical sense, a knowledge component is implemented as an object within the testbed environment, thus, it must be designed in such a way that it will use message passing as its means of communication with other knowledge components. In addition, the knowledge component will have a private set of data which can be stored as a part of its state when it is not active. The net effect of partitioning the intelligence data fusion problem solvers into knowledge components in the object-oriented environment is that the knowledge components can serve as the basic building blocks for building a data fusion architecture. Thus, a prototype engineer can insert, delete, or interchange the knowledge components in a modular manner; this capability is magnified further by the fact that the data flow among knowledge components is easily manipulated due to the message passing approach to communication.

The knowledge components work on different parts of the same global problem (e.g. situation assessment of the battlefield). Thus, the goal is to develop the knowledge components in such a way that their actions are interrelated and cooperative in nature. Communication among knowledge components is handled through message passing as provided by the object oriented environment. The messages are time-stamped, thus the synchronization of actions among knowledge components does not have to be programmed by the application developer,

rather, it is handled by the object-oriented shell.

In addition to communicating directly with one another through message passing, the knowledge components may also communicate indirectly by sending messages to a blackboard defined by the prototype developer. Hence, the Knowledge Workbench provides two means of communication for the knowledge components: (1) direct message passing, and (2) the blackboard.

Inference Engine

The testbed also includes an inference engine which can be used to process the rule-based knowledge components. The typical expert system has three components: a rule base, a fact base and an inference engine. Expert systems shells usually provide the entire context from which a system is run, with non-rule based procedures being callable by the expert system, but not invocable apart from it.

The approach in the Knowledge Workbench is to make the expert system tools available without subsuming the entire computational context. In this way the rule-based components blend in with the other programming paradigms in the system. Because of this philosophy of making the inference engine a resource instead of a context controlling driver, the rule bases are stored in knowledge components and fact bases are loaded onto a blackboard. Both of these items are implemented as objects from the system's point of view, hence the communication by message passing is enforced, thereby allowing us to break apart the expert system into component parts. This permits us to create a mixed computing paradigm to combine the aspects of rule-based computing with numeric and algorithmic processing.

The inference engine was designed to operate primarily in a forward chaining mode. This choice of inference technique fits the data-driven nature of a fusion system. A forward chaining inference engine begins with a set of facts or premises and seeks to make inferences or conclusion based on these facts. This is in contrast to a backward chaining system where the engine is given a goal or proposition and must prove that it is true based on the facts and rules which are in the knowledge base. The forward chaining approach to rule based processing provides a means of making all possible inferences about a set of sensor or correlated data thereby finding all of the identifiable battlefield entities, not just selected ones. This does not mean that it is not possible to set a higher priority on the search for particular items, for this can be done by partitioning the rules into categories by priority, type, or any other criteria. In fact, one of the features of the inference engine is the ability to partition the rules in a given knowledge component into sets which can be loaded or unloaded depending on the context. The context is monitored by meta-level rules defined by the user, and when a particular condition is detected the set of active rules

may be totally changed. Using the same approach at a knowledge component level, a message may be sent from one knowledge component to another as a result of a rule invocation. The message may contain data or else it may contain an activation message.

The inference engine handles the following rule features when checking the precondition of a rule:

- (a) AND - conjunctions of clauses
- (b) OR - disjunctions of clauses
- (c) NOT - negation of a clause
- (d) *expr* - a predicate expression to be evaluated and treated as true or false.

Once a ruled precondition is satisfied, the rule can be "fired" and some action can be taken. These actions include:

- (a) ADD - add a proposition clause to the blackboard
- (b) DELETE - remove a proposition from the blackboard
- (c) ASK - ask the uses for information
- (d) @ - make a procedure call or send a message to another knowledge component
- (e) LOAD - change the rule-base by loading new partition(s)

The individual clauses in a rule may contain pattern variables. The inference engine handles all of the variable bindings and instantiation while doing the pattern matching in the system. In addition, pattern matching functions are made more efficient through the use of a set of hash table retrieval functions. This is described in more detail in the next subsection.

Blackboard Object

A blackboard in the Knowledge Workbench is a data object. It receives input data in two different ways:

- (1) the user can load a priori data from a file onto the blackboard using a menu option on the Knowledge Workbench,
- (2) the knowledge components may send data to the blackboard - this data from the knowledge components may be correlated or preprocessed data, or it may be in the form of newly inferred information.

The blackboard is implemented using a hash table to make the storage and retrieval of information more efficient. The hash function operates on the second element of each clause, which is normally some sort of relational descriptor. For example, in the clause:

(?node1 controls ?net1),

- (a) COUNT {ENTITIES | ENTITY-RELATIONSHIP-GRAPHS | NODES}
& OPTIONAL filter
- (b) ENTITY-FILTER & OPTIONAL reset
- (c) ENTITIES {entity-id | ENTITIES | ALL-ENTITIES}
- (d) EXPLAIN function-name
- (e) LIST-FUNCTIONS
- (f) MEMBERS-OF {entity-relationship-graph
| ENTITY-RELATIONSHIP-GRAPHS
| ALL-GRAPHS | node-name | NODES
| ALL-NODES}
& OPTIONAL {ENTITIES | ALL-ENTITIES}
- (g) MEMBERSHIP-OF {entity-id | ENTITIES | ALL-ENTITIES}
- (h) GRAPH-FILTER & OPTIONAL reset
- (i) GRAPH-MEMBERSHIP-OF {entity-id | ENTITIES
| ALL-ENTITIES}
- (j) GRAPH {graph-name | GRAPHS | ALL-GRAPHS}
- (k) GRAPH-RELATED-TO {node-name | NODES | ALL-NODES}
- (l) NODE-FILTER & OPTIONAL reset
- (m) NODE-MEMBERSHIP-OF {entity-id | ENTITIES
| ALL-ENTITIES}
- (n) NODES {node-name | NODES | ALL-NODES}
- (o) NODES-RELATED-TO {graph-name | GRAPHS | ALL-GRAPHS}
- (p) QUIT
- (q) SHOW-FILTERS

Figure 1. Order of Battle Database Query Language

KNOWLEDGE WORKBENCH

SYSTEM OPERATION	DATA ACCESS
1 - PROCESS NEXT MESSAGE BLOCK NEXT BLOCK: [090900] LAST BLOCK: [091230]	20 - QUERY ORDER OF BATTLE DB 21 - QUERY BLACKBOARD
2 - RUN EXPERT SYSTEM SHELL	22 - DISPLAY INPUT MESSAGES 23 - DISPLAY OUTPUT MESSAGES 24 - DISPLAY FEBA TRACE 25 - DISPLAY EXPERT SYSTEM STATS
ENGINEERING/SYSTEM PARAMETERS	OUTPUT SWITCHES
10 - SET MESSAGE BLOCK TIME SIZE [30 min] 11 - ENTER FEBA TRACE : [AS OF 090900] 12 - EDIT RULE BASE 13 - LOAD FILE 14 - ACTIVATE RULE BASE: "TWNKB" 15 - ACTIVATE BLACKBOARD: "AAA" 16 - LOAD DATA ONTO BLACKBOARD: LOADED 17 - RESET RULES/BLACKBOARD	30 - APPLICATION MESSAGES: OFF 31 - OBJECT ACTIVATION: OFF 32 - DEBUG MODE: OFF 33 - SESSION OUTPUT TO FILE: OFF 34 - DATA ACCESS OUTPUT TO: SCREEN
	q - QUIT

ENTER CHOICE:

Figure 2. Knowledge Workbench User Interface

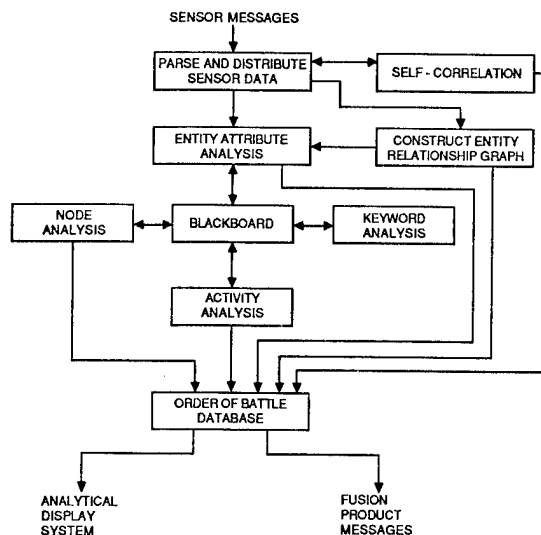


Figure 3. Fusion Prototype

"?node1" and "?net1" are pattern variables and "controls" is the relational descriptor for the two variables. It should also be noted that only instantiated clauses are stored on the blackboard, hence, the two pattern variables "?node1" and "?net1" could be replaced by actual node and net identifiers. When there are multiple instances of clauses with the same relational descriptors, a list is formed; the inference engine accesses a clause list from the blackboard, and it searches the list for a match with the current precondition clause. Hence,

(?entity part-of ?node)

matches:

(id-1000 part-of node-10)

where ?entity is bound to id 1000, and ?node is bound to node-10.

Message Processing

Sensor data is sent to a data fusion prototype in the testbed from a file of formatted sensor messages. In the application we built using the Knowledge Workbench, the file of formatted sensor messages was broken into time interval blocks as specified by the user with the menu selection for block size. Once the block size is specified, the messages from that block can be read into the application and parsed for use by the knowledge components. Thus, it is possible to handle varying sized blocks of messages as long as they are sent to the input file.

Output message processing is handled in a similar way to input message processing. Once there is sufficient information about a battlefield entity or node in the order of battle database, a formatted message is generated containing information about the entity, its location, and its activity. The knowledge component which formats the order of battle data also stores the message to a file.

Order of Battle Database Object

Another special type of data object in the Knowledge Workbench is the Order of Battle Database. This database is initialized with a priori data about the opposing force's strength and disposition profile; it is subsequently populated by the knowledge components responsible for identifying entities, entity attributes, entity relationships, and nodes. After each cycle in which a block of data messages has been processed, the user can make queries on the Order of Battle database using a stylized query language. See Figure 1 for a list of the queries possible in the system.

Testbed Environment Switches

The Knowledge Workbench user interface provides a number of switches and options for running, testing, and debugging a prototype.

The testbed can be used in several different modes. As a prototype developer, the user can edit a rule-based knowledge component and run it separately from a full prototype simply by selecting the menu options to edit the rule base, activate a rule-base, activate a blackboard, load data onto a blackboard, and run the expert system shell. Hence, it is possible to use the Knowledge Workbench to perform knowledge engineering activities on a rule-based knowledge component.

When debugging a prototype there are a number of different levels of output from the system ranging from no output at all to a step-by-step processing of the data. When interested in the knowledge component application, the user may toggle application messages on and off. When the focus is on monitoring the data flow, the user has two options: (1) show the activation of the different knowledge component objects, or (2) actually step through the prototype cycle a message at a time. In the step mode the user can view the message contents as well as the objects state.

Finally, the testbed provides the means to view the various database objects in the prototype as well as the input and output messages and the expert system statistics.

User Interface

The interface to the testbed is shown in Figure 2. The interface was developed to allow the user to operate the testbed in two different modes: (1) Knowledge Engineering and (2) System Operation. In the knowledge engineering mode the user can choose to edit the rule base, load data onto the blackboard, and test the rules by running the expert system shell separate from the prototype. Prior to normal system operation, the user has the ability to choose the next block of formatted messages to be read and processed by the system. Once the user starts the prototype it will process a message block until all knowledge components have completed their activations.

AN INTELLIGENCE DATA FUSION PROTOTYPE

An example of an actual fusion prototype architecture and its data flow is shown in Figure 3. This prototype was implemented using the Knowledge Workbench and currently runs in Lisp on a uniprocessor system.

Problem Decomposition

The prototype was created with a set of knowledge components to (1) process messages, (2) perform limited self-correlation, (3) construct entity relationship graphs, (4) perform analysis of entity attributes, (5) perform keyword and activity analysis, (6) and aggregate groups of entities into nodes. In addition, there were two knowledge components used to store information, namely, the blackboard and the Order of Battle database.

The input sensor data is stored as a set of formatted messages in a file; they are generated from a simulation in which the opposing forces are making a transition from one scenario to another. The user runs the prototype from the testbed interface shown in Figure 2. The messages are processed in time blocks of a user-specified size, and the new information is correlated with pre-existing knowledge about the entities being analyzed.

Data Flow

The data flow in the prototype is shown by the lines connecting the knowledge components in Figure 3. The data flow can easily be modified by merely changing the addressee in the message. One of the problems we encountered during the implementation of this prototype was keeping a definition of how each knowledge component expects its data messages to be formatted. We recommend that a message format dictionary be maintained for each knowledge component in order to minimize the problems associated with changing the data flow. In addition, each knowledge component's data dependencies should be documented since it may never be activated if it doesn't receive the data it needs.

Computational Paradigms

Included in the prototype were knowledge components which used (1) 'C' functions to do message parsing, (2) rule based systems to recognize entities, nodes, and their relationships, (3) a blackboard for cooperative problem solving, (4) a graph analysis algorithm, (5) a pattern analysis algorithm, and (6) an Order of Battle database.

FUTURE RESEARCH

There are a number of issues we wish to address in our next iteration of the Knowledge Workbench. The next obvious step is to extend the tool to operate on a set of distributed nodes instead of on a uniprocessor. The current version of the object-oriented shell simulates the Time Warp distributed operating system, hence, the knowledge sources will not have to be modified; the primary modification will be to distribute the object-oriented shell over a network of computers.

In addition to distributing the Knowledge Workbench, we are interested in the issue of creating knowledge components that are able to reason about one another in order to form coordination networks; a coordination network may be formed for the purpose of sharing knowledge, goals, skills, and plans to jointly take an action or solve a problem (Ref. 2). Such a collection of intelligent agents could eliminate the need for the prototype developer to modify the intelligence data fusion architecture since it would modify itself according to the situation.

ACKNOWLEDGEMENTS

The Knowledge Workbench was a team effort. The object-oriented shell was developed by Steve Hughes. The Order of Battle Database object and query language were developed by Mike Becker. Jeff Wike was responsible for implementing the sensor message database, the scenario for the prototype, and several knowledge components. Others who made contributions are Shig Hozaki, Richard Benninger, and Will Duquette.

This work was sponsored by the United States Army Intelligence Center and School (USAIC&S) at Ft. Huachuca, Arizona, under contract number J28-82. The views and opinions presented in this paper are entirely our own and do not necessarily represent the views of USAIC&S or the U.S. Army.

REFERENCES

1. Alan Bond, Les Gasser, "Readings in Distributed Artificial Intelligence", Morgan Kaufman (1988). (forthcoming)
2. Les Gasser, Randall W. Hill, Jr., Jon Lieb, Nicolas Rouquette, "ICE Part 1: ICE as a General Framework and Problem for Distributed AI", USC DAI Group Research Note 45, Dept. of Computer Science, USC, March, 1988.
3. Adele Goldberg, David Robson, "Smalltalk-80: The Language and Its Implementation", Addison-Wesley, Reading, Massachusetts (1987).
4. Lee D. Erman, Frederick Hates-Roth, Victor R. Lesser, D. Raj Reddy, "The Hearsay II Speech-Understanding System: Integrating Knowledge to Resolve Uncertainty", Computing Surveys, Vol. 12, No. 2, June 1980.
5. Victor Lesser, Daniel D. Corkill, "The Distributed Vehicle Monitoring Testbed: A Tool for Investigating Problem Solving Networks", AI Magazine, Fall 1983.
6. David Jefferson, H. Sowrizral, "Fast Concurrent Processing Using the Time Warp Mechanism", SCG Conferences on Distributed Simulation, San Diego, January 1985.
7. Reid G. Smith, "A Framework for Distributed Problem Solving", UMI Research Press, Ann Arbor, Michigan, 1981.
8. Mark Stefik, Daniel G. Bobrow, "Object-Oriented Programming: Themes and Variations", AI Magazine, Winter 1986, pp. 40-62.

Integrating Plans and Scripts: An Expert System for Indications and Warning¹

John W. Benoit
Edward J. Dombroski
Pamela W. Jordan
Sharon J. Laskowski

The MITRE Corporation
C³I Artificial Intelligence Center
7525 Colshire Drive
McLean, Virginia 22102

An expert system for monitoring complex activity in order to predict future actions of intelligent agents requires knowledge that is difficult to capture in standard rule-based representations. This paper describes the artificial intelligence methodology we have developed for plan recognition that leads to the design of such an expert system decision aid. The application domain motivating this research is tactical battle management where an expert system can aid intelligence analysts in discovering trends on the battlefield more quickly and accurately. However, this research can be applied to any system that is monitoring a volatile environment with incomplete and/or misleading information.

To predict actions, plausible goals and plans must be determined based on observed events and the order in which the events have occurred while taking into consideration that incoming information is often sparse or deceptive. Our research combines knowledge in script-like format and artificial intelligence planning techniques to formulate a set of hypotheses of the adversary's actions. We have focused on integrating a script-based goal detector with a plan recognizer to generate feasible plans. By specifying tactical and doctrinal sequences of events, the script representation enables the generation of a set of hypotheses about the adversary's goals. Observed events are compared to stereotypical sequences of events using flexible temporal matching techniques that we have developed. The hierarchical plan recognizer generates plans from this set of hypotheses using information about the particular situation. By limiting the number of hypotheses through script matching, the planning becomes computationally tractable. The multiple plans are generated and organized using assumption-based truth maintenance techniques tailored for use in the generation and expansion of partial plans.

The resulting expert system, IPS, can suggest several possible stereotypical goals as explanations of the sequences of events unfolding and, then, detail the plan steps that the adversary must execute in order to obtain these goals. The key contributions of this research are the combination of script and plan knowledge representations in the context of an expert system, the temporal reasoning applied to matching observed events against stereotypical scripts, and the representation of plan alternatives in a hierarchical planner. IPS itself suggests ways in which expert system technology can aid the human analyst at many levels of problem solving.

Integrating Plans and Scripts: An Expert System for Indications and Warning¹

John W. Benoit
Edward J. Dombroski
Pamela W. Jordan
Sharon J. Laskowski

The MITRE Corporation
C³I Artificial Intelligence Center
7525 Colshire Drive
McLean, Virginia 22102

Introduction

This paper describes our current research and development of artificial intelligence techniques and structures required to build an expert system decision aid that can predict the goals and actions of various intelligent actors. The application domain in which these techniques are being developed is tactical battle management where such an expert system can aid intelligence analysts in determining trends on the battlefield. However, this research, when complete, can be applied to any system that is monitoring a volatile environment with incomplete and/or misleading data.

The main objective of this project is to demonstrate that the use of artificial intelligence techniques based on *scripts* and *planning* will greatly assist intelligence analysts in the detection and identification of actions evolving over time. A method for the early detection of goals is to monitor a number of continuing phenomena, for example, troop movements, and signal an alert when any of these phenomena exhibit values outside a preselected, "normal" range. These phenomena are usually called indicators. Based on an indicator alert, analysts must hypothesize possible missions from a review of past behavior, doctrine, and previous reports. In doing so, the analysts attempt to fit the reports to plausible plans for achieving the missions. From the candidate hypotheses and their associated plans, it is possible to predict events and the related intelligence indications associated with each of these hypotheses which may be compared with future reports in order to identify the most likely action.

This analysis can be viewed as a type of plan recognition. Applicable artificial intelligence techniques have been explored in other fields of AI, such as deep semantic analysis of natural language stories and automated planning. Key to the application of these mechanisms to the intelligence domain is the representation of likely events of possible crises with a knowledge structure that allows efficient matching of actual indicator results to these events. For small domains, it is conceivable to generate all possible plans and match these against the observations. However, for any domain of even

slight complexity, this leads to an intractable combinatorial explosion of plans and plan variations. This is especially true in our case where the variations are extremely large. This calls for some method of pruning the number of likely missions and plan variations before enemy plan generation is begun. The idea of a *script* representation from Schank's research in natural language processing is applicable here [SA77]. A script is a formalized, stereotypical representation of a sequence of actions oriented toward attaining some goal. In general, there are many goals and (possibly) several stereotypical methods of attaining these goals. For our development, these scripts are obtained from knowledge of the enemy's doctrine and from the experience of the intelligence experts in the domain. Actual sequences of reports are matched against these scripts in order to identify those scripts that best explain the reports. Special care in this matching is required to account for the temporal development of a crisis. The selected scripts become an estimate of the likely missions and of the plans being pursued. For Schank, the reports were sentences in a story; in our case, intelligence reports in an unfolding action.

Although script matching allows the elimination of many hypotheses, the variability and multiplicity of steps and sequences in obtaining a goal are such that a static script representation is not adequate for detailed understanding of realistic situations. For example, due to the terrain in a specific area, the ordering of troop movements and the length of time these actions take could be considerably different than what doctrine specifies. Planning knowledge must be introduced [Wil78, Wil81, SA77] to modify and expand the script hypotheses as a situation develops. Thus, by using script matching to hypothesize a reduced number of plausible goals for the apparent crisis, it becomes feasible to generate detailed *plans* for these hypothesized goals using the observed indicator values and the current situation as guides for the available options. The plans are used to predict future observable events that could discriminate among the hypothesized goals. Requests for further critical observations to refine the hypotheses can then be generated from these predictions. The reports, along with their estimated reliability, are matched with the hypothesized plans to obtain an estimate of the likelihood of

¹This research was supported by MITRE Sponsored Research (MSR).

each plan. The plan with the highest likelihood provides the best estimate of the enemy's goal and future actions.

Background and Motivation

Previously, MITRE researchers developed ANALYST [Bon81, LAB85], an expert system that is able to infer real-time situation displays from multiple sensor sources and also to process mission-oriented information requests. ANALYST answers these requests with a rule base of static critical indicators that refer to the force disposition. ANALYST is also part of a project to construct a set of cooperating expert systems called ALLIES [Bea86] designed to perform a portion of the command and control reasoning process. ALLIES includes a military operations planning expert system (OPLANNER) [BDP86a] and an object-oriented simulation of the war (Battlefield Environment Model) [Nug84]. In the context of ALLIES, it became apparent that ANALYST was not capable of, but had the potential for, in-depth analysis that would give a clearer picture of the adversary's activities and intentions. Scripted Analyst (SCAN) [LH87] was developed to monitor continuously changing situations that include missing, inaccurate and/or deceptive sensor data and to analyze multiple adversaries while at the same time preserving the desirable characteristics of a rule-based expert system. It is based on a script representation which supports a heuristic matching technique to determine which scripts best fit a set of reports.

There are few situation monitoring expert systems that have been developed for domains as volatile as our Integrating Plans and Scripts (IPS) application. In [AFFH86, AFFH87] a plan recognizer with a simple goal detector for analyzing aircraft threat is described. Blackboard architectures have been used in domains such as speech processing, but these systems do not fully address the problems that an expert system must handle in a rapidly changing environment. Fall's work [Fal86] uses a representation called a "model" similar to scripts to propagate evidence through time for situation monitoring, but without a robust interface to a rule-based expert system. The Ventilator Manager (VM) program [BS84] is an example of a MYCIN-like system with an underlying state transition model for interpreting data in an intensive care unit, but was found to be inadequate for monitoring data continuously over time. The work of Kautz and Allen [KA86] is a bottom up plan recognizer (i.e. it tries to discover all plans that contain the reports) and in our domain would be combinatorially explosive.

MITRE has been working on AI techniques for military planning applications since 1979 and developed the KBS system in 1980. KBS was applied to joint crisis response planning in support of the Office of the Joint Chiefs of Staff. The KBS system employed a powerful but difficult to engineer procedural model of planning. This inflexibility led the MITRE researchers to consider planning from a rule-based perspective. The OPLANNER system is an investigation into the use of hierarchical planning techniques controlled by rules employing constraints within and among portions of the plan. OPLANNER is more a framework than a system, in that it is composed of a network-based rule interpreter, knowledge structures and meta-planning strategies, and a small set of example planning operators tailored to two separate planning domains. It has been applied in the past to joint operations crisis response planning [BDP86a] and to a more tactical ground battle planning scenario [Bea86]. An improved system, REDPLANNER, was developed to generate the detailed plans required for IPS. While the REDPLANNER system employs the OPLANNER rule interpreter and many of the planning lessons learned during the earlier OPLANNER efforts [BDP86b], the use of DeKleer's ATMS [dK84, dK86]

concepts for maintaining knowledge about multiple plan alternatives is a major enhancement of the OPLANNER AI planning framework.

Approach

In a previous section we have discussed the motivating influence of semantic analysis of natural language. However, the analogy to natural language understanding should not be carried too far. There are several features of the intelligence plan recognition problem that are not significantly present in natural language understanding:

- Most events have a characteristic duration that is significant in the recognition of plans.
- It appears to be necessary to carry forward multiple hypotheses and to reevaluate previously processed reports.
- Large numbers of unrelated reports are received.
- The reports (not just their interpretations) have inherent associated errors and ambiguities.
- The reports are from different sensors and thus have different significance.

The approach described below was motivated by these issues.

The basic structure of the IPS system is shown in Figure 1. SCAN matches the reports against a set of static scripts in order to suggest a number of possible enemy goals. We assume multisensor correlation prior to IPS by using only processed reports; that is, the reports are of units and their activity (for example, motorized infantry moving west) not of sensor data. This simplification was necessary to limit the scope of the IPS project. SCAN reasons about event durations and time relations between events, but does not represent turning points (alternative sequences of events within a script).² When SCAN determines that a script (or scripts) has been matched, it passes on the script goal for a specific context (locations and time) and the related reports to the REDPLANNER. REDPLANNER uses the goals and reports from SCAN to generate alternative plans for attaining the goals. While the scripts are stereotypical and static, the plans generated account for the current situation and alternatives in attaining the goals. REDPLANNER is a hierarchical planner using modified procedural networks to represent the plans. In order to efficiently generate and represent many alternative plan structures, the procedural networks were augmented by some of the techniques of Assumption-based Truth Maintenance Systems (ATMS) [dK84, dK86]. The alternative plans for the goals hypothesized by SCAN are analyzed by the Plan Analyzer.

The Plan Analyzer compares the plans with each other and with the reports in order to rank them as estimates of the situation. It also searches for predicted events the can be observed and that discriminate among the plans. If these events are observed, pruning the set of hypothesized plans and goals is possible. The discriminating predicted events are passed to SCAN as cues for its monitoring and matching of incoming reports.

Development and testing of IPS is done with simulated data. Since the IPS system reasons about causal relationships among events, the simulated events must appear to be causally related. The simulation, Soviet Tactical Forces Environment Model (STFEM)³, specially constructed for IPS,

²Alternative sequences are dealt with in the planning phase.

³STFEM was designed and implemented by John R. Davidson and Russell R. Leighton of the MITRE Corporation.

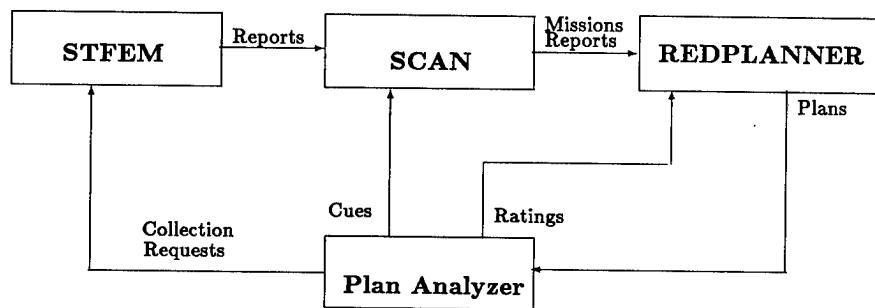


Figure 1: IPS Architecture

produces these events by using scripts to represent the simulation scenarios. These scripts (which are slightly different from the SCAN scripts) describe events in their causal relationships for expansion by the simulator.

The following sections provide additional descriptions of these four parts of the IPS system.

SCAN

From a set of reports based on behaviors generated by STFEM, SCAN recognizes and monitors trends in activities as they unfold and passes its evaluation of the current status onto the REDPLANNER for further refinement as shown in Figure 1. These reports from STFEM include information about units, their size and type, direction of movement and level of activity. It is assumed that the problem of coalescing information from multiple sources is managed at the simulation end. Unlike ANALYST, SCAN deals only with the higher level information and with scenario detection and not with data fusion. SCAN searches a set of scripts to determine those which are plausible explanations of the current activity, that is, which scripts have a strong likelihood of matching against the observed behavior.

The script knowledge base is stored as a taxonomy of indicators which describe stereotypical behavior. A SCAN script (see Figure 2, which is not necessarily indicative of true doctrine) includes a list of events called script-elements such as *increased-logistics-and-service-activities*. Time relations between script-elements are expressed by the ordering predicates. For example, *increased-reconnaissance* occurs during *increased-logistics-and-service-activities* with the former beginning approximately 60 minutes after the latter has started. The time formalisms are based on work by James Allen [All84, AH85]. Script elements may be primitive indicators called propositions (such as *increased-reconnaissance*) or they may point to other scripts in the hierarchy. The subscript *attack-with-tank-div* starts approximately 120 minutes after *increased-reconnaissance* starts and may continue after it ends. Either an *attack-with-tank-div* or an *attack-with-motorized-rifle-div* could be occurring in a frontal attack, therefore, both are monitored. *bindings* are passed to the script. They describe the context of the instantiation of a script or script-element (e.g. location and time) and the default constants such as the duration of an event. The preconditions specify entry conditions for the script to be a candidate hypothesis. They are divided into necessary and sufficient preconditions based on whether the condition is static or dynamic. For example, a frontal area is a necessary condition for a frontal attack as represented in Figure 2; while in another script, the presence of three tank battalions could be a sufficient precondition for an attack by a tank division.

```

(defscript frontal-attack
  (script-name frontal-attack)
  (script-elements
    (increased-logistics-and-service-activities
      proposition
      (>location >time (>duration infinite)) nil)
    (increased-reconnaissance
      proposition
      (>location >time (>duration 240))
      (during increased-logistics-and-service-activities 60))
    (attack-with-tank-div
      script
      (>location >time)
      (overlaps increased-reconnaissance 120))
    (attack-with-motorized-rifle-div
      script
      (>location >time)
      (overlaps increased-reconnaissance 120))))
  (bindings (>location >time))
  (necessary-preconditions (frontal-area))
  (sufficient-preconditions t)
  (script-analysis max-script-likelihood))
  
```

Figure 2: Simplified Frontal Attack Script

The first step in determining which scripts are applicable to the observed behavior is to find those scripts with satisfied pre-conditions under the current context (time and location). The underlying primitive events are extracted from the resulting subset of scripts. SCAN then applies rules about what situations contribute to the existence of the primitive events in order to determine the likelihood of each primitive event occurring. An overall likelihood is associated with the scripts based on the likelihoods of their constituent primitive events and the time relations between those events. The process of determining the script likelihood is complex because it must take into consideration the ordering and duration of events as well as missing events. Data on the events are derived from reports which are often imprecise. In addition, the activities themselves might not follow the stereotypical script exactly. To determine the overall script likelihood, SCAN assigns a likelihood to each script element (up to the one for the current time) by multiplying the old script element likelihood by the percentage of sequencing preconditions and duration information that match the reports. This likelihood can be further adjusted by weighting it according to the importance of that element in the script. Then, an average of all these likelihoods is computed to arrive at the script likelihood. If the resulting script likelihood is high, then the script is con-

sidered to be active. As time goes on, the script likelihood will either stay high enough for the script to remain active, or the likelihood will decay as script elements are missed or occur out of sequence.

The active scripts at any given point in time represent a set of hypotheses of the enemy's intentions. This list of active scripts and their associated bindings are passed on to the REDPLANNER for further consideration. REDPLANNER takes the candidate scripts and goes through a process of further refinement and generation of multiple plans. Even with a constrained set of candidate scripts, the number of possible scenarios when expanded may be very large. Therefore, the planner may send queries back to SCAN. These queries can include questions about the disposition of units in a particular location at a given time. In addition, the Plan Analyzer, which evaluates plan consistency, may suggest other hypothesized missions for SCAN to monitor. A more detailed description of SCAN may be found in [LH87].

REDPLANNER

REDPLANNER is a rule-based, hierarchical planner using constraint propagation to limit the search for plausible alternatives [Sac77, Ste81, Wil84, BDP86a]. Plans and their abstractions are represented in procedural networks [Sac77]. Procedural knowledge in the nodes of the network is represented by rules. Constraints consist of dynamic typing of the plan variables and of rules relating multiple plan variables. Figure 3 shows the hierarchical structure of a plan. In this extremely simplified example, the red mission is an **ATTACK** which has been decomposed into three phases, **ASSEMBLE**, **DEPLOY**, and **STRIKE**. In a more detailed representation, each of the sub-plans would be further decomposed into sub-plans such as **TRANSPORT**, **MANEUVER**, etc.

Alternative plans and goals are represented by incorporating the techniques of Assumption-based Truth Maintenance Systems (ATMS) [dK86] into the procedural network. Whenever the constraints cannot eliminate all but one alternative, the planning system must make choices. These choices are represented as assumptions in the ATMS structure. Alternatives are mutually inconsistent choices. Thus, a given alternative plan is described by the set of choices one believes—automatically excluding those choices inconsistent with the current beliefs. Another plan is described by a different set of choices. The rule evaluation mechanism ensures that the variables of the rules are consistent so that only inferences based on consistent sets of assumptions are made.

The function of REDPLANNER is to generate feasible plans for the hypothesized missions received from SCAN. These plans must be consistent with the reports and the current situation. SCAN will have passed on the reports used to match the script along with the script context identifications it has made. Recognizing that the reports are inaccurate and that SCAN has had to force them into a finite number of stereotypical scripts, REDPLANNER uses the reports associated with specific plan elements only as guidance in plan generation. The plans generated loosely follow red doctrine and the alternatives selected are those most closely matching the reports. In general, the reports are ambiguous and have likelihoods associated with them; the selection of plan alternatives takes these likelihoods into account. In addition to allowing flexible selection of alternatives, the use of planning allows the analysis to be constrained by the current situation; that is, the political situation, the local terrain and weather, the resources available, and the possible counter actions all affect the actions likely to occur. REDPLANNER will consider these factors in generating plan alternatives.

Even with the report and situational constraints, the number of possible plans is large. To help in limiting what

is a combinatorial explosion, not all plans are expanded completely. The resulting partial plans include significant past and near term observable events but do not include less significant details or many alternatives into the future. As reports arrive and time passes these partial plans are pruned, expanded, and modified to maintain consistency and currency.

This may be illustrated by an example. Figure 4 shows an idealized battle area roughly modeled after the Fulda Gap. The scenario in this example is a Red attack on Bad Hersfeld as shown in Figure 4. The first indication REDPLANNER has of this activity occurs during the initial movements toward the assembly area when SCAN suggests the possibility of an attack (among other missions such as force replacement) in the general Fulda-Bad Hersfeld area. This suggestion is based on reports that indicate a western movement of combat forces in the area of Werra as illustrated in Figure 5. At this point, REDPLANNER could generate detailed plans for an attack. Since there are many possible objectives, deployments, assembly areas, etc., many alternative plans would be developed. This is inefficient, confusing, and unnecessary. With this paucity of information, REDPLANNER generates only a partial plan based on movements in general areas and without detailed unit planning. This plan is like that represented in Figure 3 in that it considers only mass troop movements between general areas. Figure 6 shows the areas considered. The abstract plan estimates duration and general directions of the movements. As additional reports arrive and a better indication of the Red activity is obtained, more specific objectives and movements are planned.

Plan Analyzer

The basic function of Plan Analyzer is to identify the most likely mission and the plan being executed to achieve that mission. In order to do this, it evaluates the generated plans for consistency with the reports, for general feasibility, for susceptibility to counteraction, for consistency with the global situation, etc. As new reports arrive, they are used for plan re-evaluation. The analysis of the alternative plans supplied by REDPLANNER are closely integrated with the actual plan development since the analysis may suggest additional alternatives or additional expansion of some aspect of a plan in order to compare with new reports or to compare plans. Another function of Plan Analyzer is the prediction of observable, discriminating events to assist in the evaluation and ultimate pruning of plans. To do this, plan differences need be discovered and knowledge of sensor capabilities is required. These predictions are passed to SCAN so that it will monitor the reports for related information. Later development of Plan Analyzer will include passing collection requests to the simulated sensors.

A powerful future function of Plan Analyzer is to recognize repeated plans that do not have associated SCAN scripts. This means that a previously unknown method of achieving some goal has been discovered. This could detect doctrinal changes as well as omissions and errors in the script definitions. Having made such a recognition, the next step is to generate a new script for SCAN. This requires extensive knowledge of script structure and semantics.

Since the Plan Analyzer must deal with the procedural network representation of the alternative plans, it will be implemented as an asynchronous phase of the planning system. As mentioned above, the planning system is rule driven. It will be necessary to add a network matching capability to this system in order to perform the plan analysis.

STFEM

STFEM is an object-oriented, script-driven, scenario generation tool that provides a directed set of sensor reports to

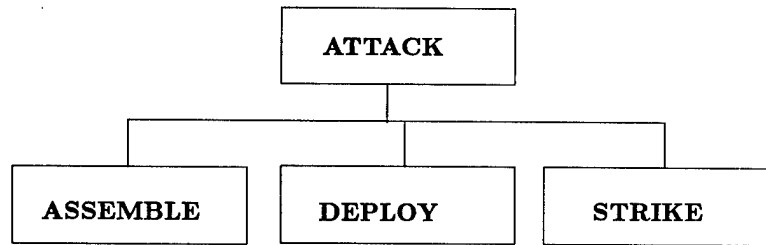


Figure 3: A Simple Plan

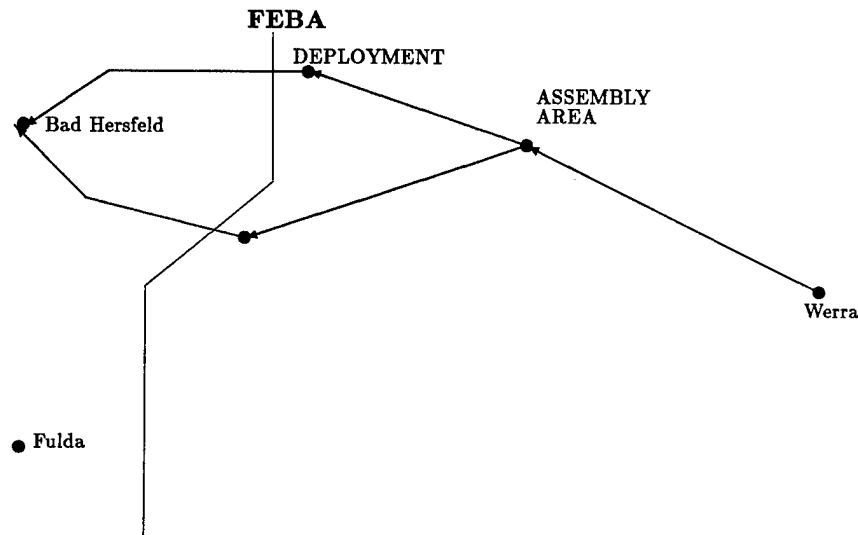


Figure 4: A Simple Red Action

feed the IPS intelligence support system. It builds on work at MITRE in advanced behavioral simulations [Nug84, MK82]. Simulation software tends to be complex and highly focused for a particular domain application. When used to supply ground truth for intelligence support systems, such simulations are often difficult to alter or extend. STFEM was developed as a scenario generation tool which, while providing a suitably rich environment for a variety of intelligence functions, would also allow users to create these scenarios in a flexible way. User interaction with STFEM is based on defining and instantiating a series of spacial templates and procedural scripts which form the basis for the scenario. The current situation is an attack in the Fulda Gap area of Germany. In this case STFEM begins with a set of default attack scenarios. Throughout the scenario definition process, the user can simply choose a certain default configuration, alter a given configuration, or define a new scenario within the structure of a very high level attack script.

To create a STFEM scenario for the current situation, a user first chooses one from a set of available attack scripts. Each attack script describes a road march from a garrison area through assembly and deployment areas to a target that is selected by the user. STFEM currently supports pincer, envelopment, and flank attack scenarios. While each script defines basic deployment and attack plans and configurations,

STFEM allows the user to choose the route and tactical area locations as well as the military units that compose the attack force. Only these three scenarios are currently supported; however, the STFEM scripts can easily be augmented to support any number of scenarios. These scenarios, in turn, can be used to provide a rich set of sensor reports to other intelligence support systems.

The scenarios are single-sided and concentrate on unit movement and communication. STFEM generates observables in an aggregated sense. Rather than simulate each event such as a radar emission or communications message, it models the level of activity over time for each of five observable phenomena. These are: communications activity, radar activity, motion, indirect fire, and optical visibility. Each unit involved in a particular phase of the operation is assigned a level representing the rate of movement, communications, radar activity, etc., for that time period. These histograms of activity are carried in the object structures for each actor as the actor pursues the assigned military goals. They are contextually derived and form one of the bases that SCAN and REDPLANNER use in computing the likelihoods of events.

STFEM sensors are tasked by regions that act as areas of focus. In this way we can task the system to collect more intelligence signals in a particular area or "fly" an imagery mission in another area without individually modelling the instances

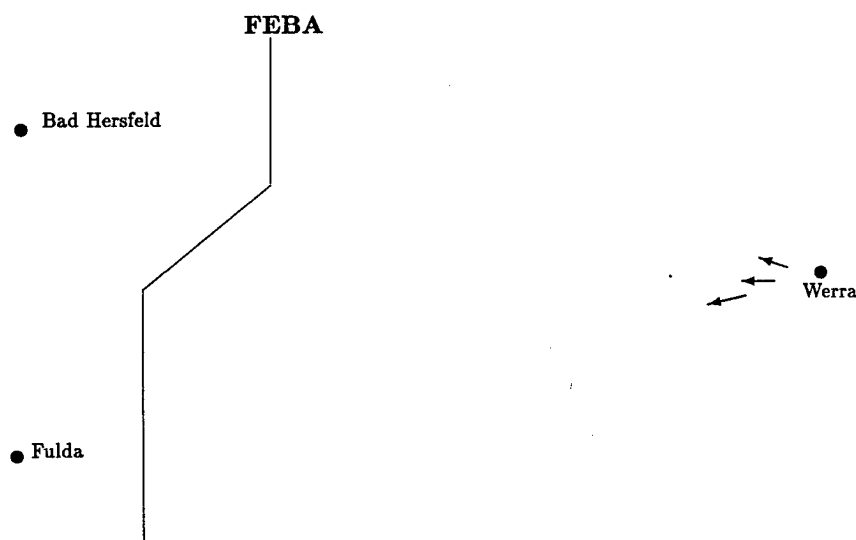


Figure 5: Initial Reports

of those sensors. Each region has a level of "acuity" for each of its observable classes. This acuity is the probability and accuracy with which the report is made. The acuity associated with each sensor is also used to provide an error measure that may be used as an indication of the confidence in the report. This is an acknowledged approximation of tactical sensor suites that would not be appropriate in other contexts, but serves well in the IPS system.

Accomplishments

The IPS project is a little more than half complete. An initial demonstration system containing all the parts shown in Figure 1 has been built, but the full functionality described above is not yet available.

The scenarios used for the STFEM simulation are oriented towards ground activities and include flank, frontal and pincer attacks. These unclassified scenarios are suitable for the proof of concepts that will have a wider domain of applicability. The STFEM tools are completed, but more and richer scenarios will be needed for future research.

SCAN has new scripts that stereotypically model several missions, including those of the simulated scenarios. Simple spatial template matching is used to detect force deployment. When SCAN receives some reports from the simulated sensors and assigns one or more missions to that activity, these assignments and the sensor reports are passed on to REDPLANNER.

REDPLANNER incorporates augmented Assumption-based Truth Maintenance System (ATMS) techniques in order to facilitate the generation of alternative and contingency plans. When REDPLANNER receives hypothesized missions and sensor reports from SCAN, it generates a number of alternative plans based on the mission assignments, the sensor reports and red doctrine.

The Plan Analyzer is partially developed. It performs a limited comparison and evaluation of plans based on doctrine and new sensor input. It also predicts events, but the ability to detect events that are discriminatory is limited.

The demonstration system includes two innovative advancements of existing techniques. The first involves tempo-

ral reasoning about durations of action that are of uncertain length. Scripts express doctrinal or experiential knowledge about event timing. In practice, the time required to complete an event (such as movement) may vary considerably from action to action. This will cause the time between reports of sequential events to deviate significantly from the stereotypical value. It is insufficient to test only the sequence of events since, in general, the timing of events is critical to proper script matching. For these reasons, the temporal reasoning development included a flexible time interval matching procedure.

The second innovation was the development of a representation for alternative plans in a hierarchical planning system. Since there may be several possible missions and many ways to achieve these missions, even within the constraints of the situation and reports, it was critical that a conceptually perspicuous method for describing and comparing these plans be developed. It was also necessary to develop a computationally efficient technique for developing these plans. The techniques of ATMS, which, in effect, qualify every part of the plan representation by its plan alternative, were enhanced to provide the required representation and control.

Future Work

The number of scenarios and indicators must be expanded and the inference rules improved to demonstrate the operational feasibility of IPS. Further development of the comparison of plans and the prediction of discriminating events is required. A more formal planning language is necessary to enhance plan comparison through the use of partial plans, deferred plan completion, and replanning. Mechanisms for the feedback of new scripts as well as sensor report predictions must be included to increase the sensitivity and selectivity of SCAN.

References

- [AFFH86] J. Azarewicz, G. Fala, R. Fink, and C. Heithecker. Plan recognition for airborne tactical decision-making. In *AAAI*, pages 805-811, 1986.

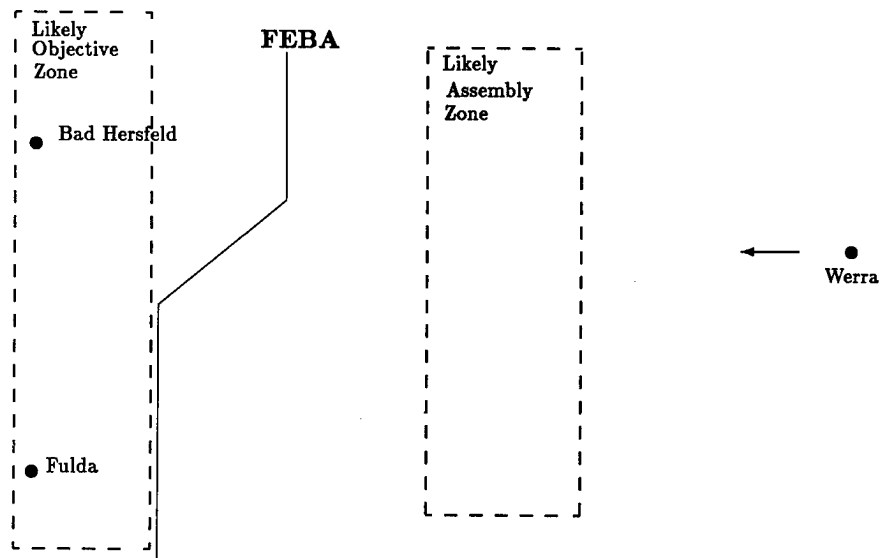


Figure 6: Abstract Attack Plan

- [AFHH87] J. Azarewicz, G. Fala, I. Hayslip, and C. Heithecker. Mult-agent plan recognition in an adversarial domain. In *ESIG*, pages 188–193, 1987.
- [AH85] J. Allen and P. J. Hayes. A common sense theory of time. In *IJCAI*, pages 528–531, 1985.
- [All84] J. Allen. Towards a general theory of actions and time. *Artificial Intelligence*, 23:123–154, 1984.
- [BDP86a] J. W. Benoit, J. R. Davidson, and E. G. Powell. Artificial intelligence in joint deployment planning. In *Artificial Intelligence for Military Applications*, ORSA, Arlington, VA, October 1986.
- [BDP86b] J. W. Benoit, J. R. Davidson, and E. G. Powell. Lessons learned developing a planning system. In *Proceedings of the 1st International Conference on Applications of Artificial Intelligence in Engineering Problems*, Springer-Verlag, Southampton University, U. K., April 1986.
- [Ben86] J. W. Benoit, et al. An experiment in cooperating expert systems for command and control. In *Expert Systems in Government Conference*, pages 372–380, October 1986.
- [Bon81] R. P. Bonasso. An expert system for processing sensor returns. In *The First Army Conference on Knowledge-Based Systems for CSI*, pages 219–245, Ft. Leavenworth, November 1981.
- [BS84] B. G. Buchanan and E. H. Shortliffe. *Rule-Based Expert Systems*. Addison-Wesley Publishing Co, 1984.
- [dK84] Johan de Kleer. Choices without backtracking. In *Proceedings Fourth National Conference on Artificial Intelligence*, pages 79–85, Austin, TX, 1984.
- [dK86] Johan de Kleer. An assumption-based TMS. *Artificial Intelligence*, 28:127–162, 1986.
- [Fal86] T. C. Fall. Evidential reasoning with temporal aspects. In *AAAI*, pages 891–895, 1986.
- [KA86] H. A. Kautz and J. F. Allen. Generalized plan recognition. In *AAAI*, pages 32–37, 1986.
- [LAB85] S. J. Laskowski, H. J. Antonisse, and R. P. Bonasso. ANALYST II: a knowledge-based intelligence support system. In *Second IEEE Conference on Artificial Intelligence Applications*, pages 552–563, December 1985.
- [LH87] S. J. Laskowski and E. J. Hofmann. Script-based reasoning for situation monitoring. In *AAAI*, 1987.
- [MK82] D. McArthur and P. Klahr. *The ROSS Language Manual*. Technical Report N-1884-AF, RAND Corporation, 1982.
- [Nug84] R. O. Nugent. MITRE's Battlefield Environment Model. In *ORSA/TIMS Conference*, November 1984. Working Group Presentation.
- [SA77] R. C. Schank and R. P. Abelson. *Scripts, Plans, Goals and Understanding*. Lawrence Erlbaum Associates, New York, NY, 1977.
- [Sac77] E. D. Sacerdoti. *A Structure for Plans and Behavior*. Elsevier North-Holland, Inc., 1977.
- [Ste81] M. Stefik. Planning with constraints (MOLGEN: part 1). *Artificial Intelligence*, 16:111–140, 1981.
- [Wil78] R. Wilensky. *Understanding Goal-based Stories*. PhD thesis, Yale University, 1978.
- [Wil81] R. Wilensky. PAM. In R. C. Schank and C. K. Riesbeck, editors, *Inside Computer Understanding*, chapter 7, Lawrence Erlbaum Associates, Hillsdale, NJ, 1981.
- [Wil84] D. E. Wilkins. Domain-independent planning: representation and plan generation. *Artificial Intelligence*, 22:269–301, 1984.

APPENDIX A

REGISTERED DFS-88 ATTENDEE LIST

THIS PAGE LEFT BLANK INTENTIONALLY

DFS-88 ATTENDEE LIST

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Abbott	Dr. Ned E.	E-Systems, Inc./Garland Div.	214-272-0515
Abbott	Dr. Richard J.	Lockheed Aeronautical Systems Co.	818-847-1455
Abend	Mr. Kenneth	Interspec, Inc.	215-834-1511
Adair	Mr. Sidney T.	Flam & Russell Inc.	215-674-5100
Adams	Mr. Neil J.	Loral (Defense Systems Division)	216-796-9885
Agnello	Ms. Jennie M.	D. H. Wagner, Associates	202-767-3882
Ajmera	Dr. Ramesh C.	Naval Weapons Center	619-939-3588
Albanese	Mr. Anthony J.	SENSIS Corporation	315-682-7777
Alderson	Ms. Susan L.	NOSC	619-553-4045
Alexander	Dr. Lawrence D.	General Electric	609-866-6253
Allen	Mr. Robert W.	Boeing Aerospace Co.	206-655-8544
Alter	Ms. Donna P.	E-Systems, Inc./Melpar Div.	703-352-1879
Anderson	Dr. Stephen L.	D. H. Wagner, Associates	202-767-3882
Anderson	Mr. Kenneth S.	Locus, Inc.	703-960-1000
Annen	Mr. Robert	ITT Avionics Division	201-284-2032
Annen	Mr. Robert P.	ITT Avionics	
Antonik	Mr. Joseph	Rome Air Development Center	315-330-3175
Antony	Mr. Richard T.	USA Ctr. for Signals Warfare	703-347-6445
Armstrong	Mr. John E.	GTE-Gov't. Systems Corp.	415-966-2722
Askin	Dr. Kurt	NRL	202-767-5966
Attili	Mr. Michael A.	M/A-COM, Inc.	617-272-3000
Babcock	Dr. James H.	The MITRE Corporation	703-883-7033
Baer	Mr. Richard T.	ODASD(I)	202-695-4224
Bailey	Mr. Robert E.	UNISYS	703-620-7287
Baran	Mr. Robert H.	Naval Surface Warfare Center	301-394-1961
Barnes	Dr. Christopher	Los Alamos National Laboratory	505-667-4370
Barnett	Mr. Paul G.	Johns Hopkins Univ.	301-953-5000
Barr	Mr. John K.	The MITRE Corp.	617-377-6879
Barrett	Dr. Terence W.	W. J. Schafer Associates	215-558-7900
Bartusek	Mr. Donald L.	Naval Surface Warfare Center	202-394-3177
Batchelder	Mr. Charles	Grumman Data Systems	619-224-3500
Baxter	Mr. Sam E.	General Dynamics/Fort Worth Division	817-763-4046
Beamer	Mr. Walter C.	Naval Air Development Center	215-441-2865
Beavers	CDR Michael C.	OPNAV	703-644-0122
Becker	Mr. Daniel	Naval Air Development Center	215-441-1171
Belkin	Dr. Barry	Daniel H. Wagner Assocs.	215-644-3400
Bendekovic	Mr. John P.	System Planning Corp.	703-276-6969
Benedict	Mr. Richard D.	Litton Data Systems	818-901-2048
Benites	Mr. Ernesto	Naval Weapons Center	619-939-3651
Bennett	Mr. Steven	The MITRE Corp	617-271-2501
Bent	Mr. Nathaniel E	Decision-Science Applications, Inc.	703-243-2500
Berman	Dr. Arie	Hughes Aircraft Co.	213-647-9471
Bernardin	Dr. Charles P.	Texas Instruments	214-343-7702
Bertapelle	Mr. Ken A.	McDonnell Douglas	714-896-1565
Betts	Mr. Roger S.	Analytics, Inc.	703-246-9060
Betz	Mr. John	TASC	617-942-2000
Betz	Mr. John W.	TASC	617-942-2000
Bianchi	Ms. April	Martin Marietta Info. & Comm. Sys.	303-977-0206
Billig	Mr. Lewis S.	Rand Corp	213-393-0411
Bogart	Dr. Elliot	TRW	703-876-8003
Bowen	Dr. Julius I.	Johns Hopkins University APL	301-953-5000

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Bowen	Mr. Robert D.	General Dynamics	619-592-5054
Bowman	Dr. Chris L.	Ball Systems Engineering Div.	619-457-5550
Brain	Mr. Gary G.	System Planning corp.	703-276-4725
Brandstadt	Mr. Jeffrey C.	GE Aerospace Electronics Sys. Dept.	315-793-7357
Brobst, Jr.	Mr. Donald G.	UNISYS Corporation	703-620-7846
Broman	Mr. Vincent P.	NOSC	619-553-1641
Brown	Dr. Robert R.	The Rand Corporation	213-393-0411
Brown	Mr. Charles H.	McDonnell Douglas Corp.	703-247-3514
Buckley	Mr. Peter	E-Systems, Inc./Melpar Div.	703-560-5000
Buddenberg	LCDR Rex A.	US Coast Guard Headquarters	202-267-1206
Buede	Dr. Dennis M.	Bendix/Decision Logistics	703-860-3678
Buemi	Mr. Robert P.	Daniel H. Wagner, Assoc.	215-644-3400
Buick	Mr. John R.	Loral Electronic Systems	914-964-2828
Bullen	Mr. George	Booz, Allen & Hamilton, Inc.	703-893-0040
Bunn	Mr. Roger A.	USAF ASD/RWZE	513-255-995862
Burke	Mr. H. Leonard	Naval Air Systems Command	202-692-3965
Burns	Mr. James E. S.	Litton Data Systems	818-902-4384
Butz	Mr. Will A.	E-Systems	703-352-0300
Campo	Mr. Charles P.	TASC	617-942-2000
Cannell	Mr. Marshall H.	MITRE Corp	617-271-7082
Carson	Mr. Michael H.	GTE Government Systems	301-738-8958
Carson, Jr.	Mr. Robert R.	Norden Systems	516-845-2498
Case	Mr. Gary R.	Lockheed Missiles & Space Co., Inc.	703-841-5022
Chan	Mr. Chi K.	M.I.T. Lincoln Laboratory	617-981-7934
Chapin, Jr.	CAPT Robert W.	HQ COMSPAWARSYSCOM	202-692-1095
Chapman	Mr. Howard M.	USMC Deputy Program Mgr. Intelligence	703-640-2247
Chin	Dr. Leonard	MAGNAVOX/GAC	215-699-8851
Chu	Dr. Wesley W.	W. W. Chu Associates	213-454-9314
Chu	Ms. Cecelia H.	NORDA	601-688-4790
Chuang	Mr. Ping C.	NSWC/WO	202-394-1237
Chubb	Mr. Douglas W.	US Army Center for Signals Warfare	703-347-6445
Chung	Mr. Charles W.	Rockwell International	213-420-4910
Clark	Mr. Ronald C.	ITT Gilfillan	818-988-2600
Clark	Mr. Terrence K.	Pacific Missile Test Center	805-989-9436
Connor	Mr. Edward D.	Corvus Corporation	703-893-0228
Converse, Jr.	Mr. Leonard E.	RADC/IRRA	315-330-3175
Cook	CW2 Lawrence R.	Joint Tactical Fusion Program	703-556-2983
Corbeil	Mr. Allan F.	Technology Service Corporation	203-268-1249
Corbet	Mr. Dan	TASC	703-558-7458
Cronin	Mr. Terence M.	U.S. Army Cecom Ctr. for Signals Warfare	703-347-6430
Crowder	Mr. Kenneth	Boeing Company	206-865-4587
Crowder	Mr. Kenneth V.	Gov't. Info. Sys/Boeing Computer Science	206-865-4587
Cummings	Mr. C. Alan	CAS, Inc	205-895-8636
Daly	Mr. Daniel A.	Litton Data Systems	818-902-4404
Davidson	Mr. John R.	The MITRE Corp.	703-883-6041
Davis	Mr. Edison M.	NADC	215-441-2514
Davis	Ms. Laura C.	Naval Research Lab.	202-767-2877
DeFoe	Mr. Douglas N.	Textron Defense Systems	617-381-4766
DeRieux	Mr. Thomas L.	Naval Surface Warfare Center	703-663-8763
DeWitt	Mr. Ronald N.	Pacific-Sierra Research Corp.	703-527-4975
Desai	Dr. Mukund N.	C. S. Draper Laboratory, Inc.	617-258-2224

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
DiPalma	Mr. Robert F.	ESY-Ctr. for Advanced Planning/Analysis	703-352-0300
Dick	Dr. Stephen G.	Hazeltine Corp.	516-261-7000
Dickinson	Dr. Philip C.	E-Systems/CAPA	703-352-0300
Diggs	Mr. James F.	Naval Research Lab.	202-767-0489
Dishman	Mr. David R.	SAIC	619-546-6237
Dooley	Mr. Richard P.	Martin Marietta Electronic Systems	305-356-6761
Dotts	Mr. Wilfred R.	Pacific Missile Test Center	516-953-3681
Drillings	Mr. Michael	Army Research Institute-BR	202-274-5572
Drummond	Dr. Oliver E.	Hughes Aircraft Co.	213-616-2624
Dunham	Lt Col Alan D.	DARPA	202-694-5738
Duniho	Mr. Michael A.	NSA	301-688-7608
Dushman	Mr. Allan	Dynamics Research Corp.	617-475-9090
Egan	Dr. John T.	Naval Research Laboratory	202-767-0762
Ehlers	Dr. David H.	The MITRE Corp.	206-655-9910
Eiserman	Mr. Gary K.	Decision-Science Applications, Inc.	703-243-2500
Emmerman	Dr. Philip J.	Harry Diamond Labs	202-394-3000
Esfandiari	Dr. Pashang	NSWC/WOL	202-394-1378
Evans	Dr. Robert H.	Naval Research Laboratory	202-767-3569
Everett	Mr. James T.	Johns Hopkins University APL	301-792-5000
Fagarasan	Dr. John T.	Hughes Aircraft Co.	714-732-0295
Faison	Mr. Joseph C.	AFWAL/CDJA	513-255-9335
Falkner, Jr.	Mr. Joe S.	COLSA, Inc.	205-830-5412
Feldman	Ms. Susan J.	Hughes Aircraft Co.	714-732-0545
Felton	Mr. Mark A.	Emerson Electric Co.	314-553-2805
Ferreira	Mr. David M.	SPAWARS	202-692-4510
Figueroa	Lt Col Robert C	Intelligence Community Staff	202-376-5560
Filippelli	Mr. Lawrence J.	Ball Systems Engineering Div.	703-528-3337
Fiorentino	Mr. Joseph S.	General Dynamics Pomona Division	714-868-1000
Fitchek	Dr. Joseph J.	Naval Ocean Systems Ctr.	619-553-2486
Fitz	Mr. Michael J.	PACMISTESTCTR	805-989-3706
Florence	Mr. Gerald P.	Unisys Corporation	703-620-7226
Fong	Mr. Franklin M	Hughes Aircraft Co.	213-647-9464
Fong	Mr. Robert C.	TRW Defense Systems Group	213-217-6249
Foote	Dr. Austin L.	Northrop Advanced Systems Div.	213-948-6534
Fowler	Mr. Thomas J.	Wright Patterson AFB	513-255-6611
Fox	Mr. Gregory	TRW	703-876-8014
Freeman	Dr. Raymond F.	Joint Tactical Fusion PMO (HQDA)	703-556-3155
Freyer	Mr. Gustav J.	COLSA, Inc.	915-779-5899
Friel	Ms. Patricia	RCA - Electronic Sys. Div.	609-722-4257
Gabriel	Mr. John R.	Argonne Nat'l. Lab.	312-972-7240
Gates	Mr. Douglas	The MITRE Corporation	703-883-5948
Gee	Dr. Sherman	Office Naval Technology	202-696-4791
Gehman	Mr. P. Robert	General Electric/Aerospace	215-354-1074
Geist	Dr. John M.	Harris Corp. - GSS	305-727-6041
George	Mr. Troy H.	UNISYS Corp.	703-558-7279
Gerber	Mr. Mark A.	ALPHATECH, Inc.	617-273-3388
Gibbons	Dr. Gregory D.	Systems Control Tech., Inc	415-494-2233
Gibson	Mr. Richard L.	Loral Defense Systems	602-925-6548
Gibson	Mr. Richard L.	Loral Defense Systems	602-925-6548
Givens	LT Andrew G.	US Coast Guard	609-729-8953
Godfrey	Mr. Scott L.	FMC Corporation	703-663-9291

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Gold	Mr. Steven B.	General Electric	215-354-1262
Goodman	Dr. Irwin R.	Naval Ocean Systems Ctr.	619-553-4014
Goroff	Mr. Spencer	UNISYS Corporation	516-574-3467
Grabowski	Dr. Martha R.	GE Tactical Systems Dept.	413-494-7606
Greer	CDR Richard E.	Space & Naval Warfare Systems Command	202-692-8974
Grubbs	Ms. Linda L.	Lockheed Aeronautical Systems Co.	818-847-1131
Gully	Dr. Sol W.	ALPHATECH, Inc	617-273-3388
Hafner	Dr. Arnold N.	Computer Sciences Corp.	619-225-8401
Hagaman	Mr. Harry T.	HRB-Singer, Inc.	703-553-6340
Hall	Mr. David L.	HRB-Singer, Inc.	814-238-4311
Hargis	Mr. Gerald	Department of Defense	301-688-6501
Harmon	BG William E.	Joint Tactical Fusion PMO (HQDA)	703-556-2930
Harmon	Mr. Kenneth W.	Naval Electronic Sys. Eng. Activity	301-862-8250
Hart	Mr. Leo A.	The MITRE Corporation	617-271-3715
Hart	Mr. Thomas M.	MITRE Corporation	617-271-8503
Hart	Mr. William A.	National Security Agency/CSS	301-688-6511
Harvey	Mr. Phillip I.	Lockheed Aeronautical Systems Co.	818-847-2750
Hassett	Mr. Daniel F.	Martin-Marietta Aerospace/Naval Systems	301-682-0027
Hatleberg	Dr. Clancy	Science Applications Int'l. Corp.	619-458-4932
Haug	Dr. Anton J.	Martin Marietta Aero & Naval Systems	301-682-1665
Hayes	Dr. James C.	COLSA, Inc.	205-830-5412
Hearn	Mr. Daniel L.	Consultant - Hughes Aircraft Co.	714-732-7791
Heath	Mr. Oliver N.	US Army Aviation Center	205-255-3973
Hecht	Dr. Herbert	SoHar, Inc.	213-935-7039
Heinz	Lt Gen Edward J	Director, Intelligence Community Staff	202-376-5610
Heithecker	Mr. Christof H.	Naval Air Development Center	215-441-3731
Herting	CAPT Thomas L.	National Computer Security Center	301-859-4450
Higa	Mr. Stan J.	Naval Weapons Center	619-939-3651
Hilbert	Mr. Donald E.	Analytical Systems Engineering Corp.	703-892-6000
Hilga	Mr. Stan J.	Naval Weapons Center	619-939-3651
Hofmann	Dr. Mark A.	DoD, US Army Human Engineering Lab	301-278-5804
Hofmann	Mr. James B.	NRL	202-767-3919
Holliday	Ms. Victoria M.	TRW	703-876-8381
Holt	Mr. James T.	Air Force Studies Analyses	202-695-5387
Houston-Ludlam	Mr. Mark D.	Frontier Technologies, Inc.	301-266-8244
Hovey, Jr.	Mr. Herbert S.	US Army Center for Signals Warfare	703-347-6594
Hud	Ms. Vicki	BBN Lab	703-284-4600
Hughes	Mr. Thomas F.	General Dynamics	817-777-8285
Humphrey	Mr. Clyde L.	Westinghouse Electric Corp.	301-765-9965
Hunigan	CAPT Kirk A.	ASD/VFAC, WPAFB	513-255-3816
Hunt	Mr. Richard V.	Pacific Missile Test Center	805-989-9586
Hunter	Ms. Kathie A.	Magnavox Electronic Systems Co.	219-429-6001
Huo	Ms. Vicki	BBN Lab	703-284-4600
Jenkins	Mr. Karl B.	Communications Systems Tech., Inc.	301-381-5080
Johnson	Mr. Joseph A.	USMC	
Jost	CAPT Randy J.	AFWAL/CDJ	513-255-9335
Jung	Ms. Natalie F.	General Dynamics Corp.	714-868-4761
Kadar	Dr. Ivan	Grumman Aircraft Sys.	516-575-5359
Kaisler	Mr. Stephen H.	DARPA	202-694-1703
Kampe	Mr. Thomas W.	Norden Systems	203-385-5746
Kaskowitz	Mr. David H.	Tiburon Systems, Inc.	408-371-9400

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Kelly	Mr. Robert J.	Allied-Signal, Inc., Communications Div	301-583-4473
Kelly	Mr. William T.	Analytics Suite 481	703-359-2850
Kessler	Mr. Keith J.	LORAL Electronic Systems	914-968-2500
Kessler	Mr. Otto	NADC	215-441-1569
Kim	Mr. Kwang H.	The MITRE Corp.	617-271-2501
King	Mr. John J.	Unisys Corp.	516-574-5423
Kirshenbaum	Capt. Jason	HQ USAF (Air Staff	202-695-4561
Kissinger	Lt Col Don F.	HQ USAF/INXXA	202-695-4561
Knuff	Mr. James P.	Logicon	619-455-1330
Kolanek	Dr. James C.	Raytheon Co.	805-967-5511
Kral	LCDR Theodore	SPAWARS	202-692-3966
Krueger	Mr. Karl H.	NSWC	703-663-8906
Kuen	Mr. Jeffrey S.	Pacific Missile Test Center	805-989-8977
Kurien	Dr. Thomas	ALPHATECH, Inc.	617-273-3388
Lagana	Mr. Richard A.	National Security Agency	301-859-4823
Lamar (Ret)	BGen Kirby	AFCEA	703-631-6235
Lamar USA (Ret.)	BG Kirby	AFCEA	703-631-6235
Lampe	Mr. Douglas H.	TCI	415-795-5322
Lancaster	Dr. James E.	McDonnell Douglas Astronautics Co.	714-896-4811
Lane	Mr. Patrick J.	E-Systems, Inc.	703-560-5000
Lange	Mr. David W.	DoD/NSA	301-688-7608
Lange	Mr. David W.	National Security Agency	301-688-7608
Larson, USAF (Ret)	Maj Gen Doyle E	Larson Electronics, Inc.	612-890-9140
Laskowski	Dr. Sharon J.	The MITRE Corporation	703-883-7929
LeVine	Dr. Donald M.	TRW Federal Systems Group/Cmd. Sup. Div	703-968-1144
LeVine	Mr. Stuart	Rockwell International AMSD	714-762-8340
Lee	Dr. Yeechun	Los Alamos Nat'l Laboratory	505-667-8715
Lee	Mr. Brian M.	Raytheon Co. (MSD)	617-274-3211
Lee	Mr. Chai	Lockheed Aeronautics Systems Company	805-257-5718
Lee	Mr. J. Preston	ORINCON Corp.	619-455-5530
Lehmann	Mr. Clark T.	McDonnell Douglas Astronautics Co.	714-896-1863
Lempicki	Mr. Walter P.	USAF	513-255-9958
Levy	Mr. Lawrence H.	Naval Air Systems Command	202-746-2120
Liggins, II	CAPT Martin E.	Rome Air Development Center	315-330-4431
Lightstone	Mr. Robert M.	Boeing Aerospace Co.	215-672-8520
Lindinger	Mr. Joseph L.	Naval Air Development Center	215-441-3189
Lindsay	Ms. Nancy P.	E-Systems Inc., Melpar Div.	703-385-5880
Livingston	Capt Mary E.	Joint Tactical Fusion Program Office	703-556-3173
Llinas	Dr. James	Calspan Corporation	716-632-7500
LoSecco	Mr. John F.	Synectics Corporation	315-337-3510
Lockie	Mr. Doug	Pacific Monolifics	408-732-8000
Loewenthal	Dr. Alex	Lockheed Aeronautical Sys. Co.	818-847-6842
Long	Mr. James P.	General Electric	215-531-1918
Long	Mr. James P.	General Electric Co.	215-531-1918
Lopez, Jr.	Mr. Alfredo	HQ, Electronics Security Cmd	512-925-2703
Loveland	CAPT Richard S.	Naval Operational Intelligence Center	202-763-3694
Luppino	Mr. Peter L.	OUSD(A)	202-697-1522
Maar	Dr. James R.	National Security Agency	301-859-6341
MacMeekin	Ms. Nancy E.	Naval Air Development Center	215-441-1720
Macmeekin	Ms. Nancy E.	Naval Air Development Center	215-441-1720
Maconachy	Mr. William V.	INFOSEC Awareness	301-688-8742

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Madan	Dr. Rabinder N.	Office of Naval Research	202-696-4217
Maginn	Mr. Mark T.	PAR Government Systems Corp.	315-738-0600
Mallach	Mr. Larry	Lockheed Aeronautical Systems Co.	818-847-1405
Maloney	Patricia Sue	Lockheed, Advanced Technology	512-448-5136
Mangeri	Ms. Lesli N.	Pacific-Sierra Research Corp.	703-527-4975
Mangoubi	Mr. Rami S.	C. S. Draper Laboratory, Inc.	617-258-2262
Marchette	Mr. David J.	Naval Ocean Systems Ctr.	619-553-4049
Martin	Mr. John W.	Bendix Communications Div.	301-583-4258
Masenten	Dr. Wesley K.	Advanced Avionics System Engineering	213-948-8979
Mason	Mr. Clark B.	Lockheed Aeronautical Systems Co.	818-847-4134
Matsumoto	Mr. Paul M.	ITT Gilfillan	818-988-2600
Mazzuca	Mr. Paul N. Jr.	Teleforce Associates	301-961-6555
Mazzuca, Jr.	Mr. Paul	Tele-Force Associates	301-961-6555
McCown	Ms. Laurie J.	TRW	213-297-3135
McCoy	Mr. Shawn K.	Martin-Marietta Aero & Naval Systems	301-682-1924
McDaniel	Mr. Eubert L.	HQ ASD/VFEA	513-255-4921
McEachern	Mr. Robert H.	HRB-Singer	301-459-8655
McKenney	Mr. Terry L.	Westinghouse Electric Corp.	301-765-1097
McLeod	Mr. Glenn A.	Naval Surface Warfare Ctr.	301-394-1394
McNamara	Mr. Robert W.	PRB Associates, Inc.	301-373-2360
McNeill	Dr. David L.	E-Systems	703-560-5000x2795
Mechtel	Mr. Gary H.	Westinghouse	301-765-6142
Meyer	Mr. Michael P.	Norden Systems	516-845-2596
Mieras	Dr. Harry	Raytheon Co.	617-274-4021
Milan	Mr. John M.	ITT Gilfillan	818-988-2600
Milford	Mr. John A.	E-Systems, Inc./Garland Div.	214-272-0515
Miller	Dr. Irwin	Raytheon Company	617-860-2856
Miller	Mr. Edwin H.	Sanders Associates	603-885-4030
Miller	Mr. Robert C.	Microwave Associates, Inc.	617-272-5008
Mills, Jr.	Mr. Harold W.	E-Systems, Inc.	703-352-0300
Miluski	Mr. John P.	PRB Associates, Inc.	301-673-2360
Mitchell, Jr.	Mr. George W.	US Army Ctr. for Signals Warfare	703-347-6791
Mittenthal	Dr. Lothrop	Teledyne Electronics	805-498-3621
Mitzel	Dr. Glenn E.	Johns Hopkins Univ./APL	301-953-5309
Moe	Mr. Gordon O.	Pacific-Sierra Research Corp.	703-527-4975
Moldoff	Mr. Barry	The MITRE Corp	617-271-3466
Monaghan	Mr. Paul J.	US Navy	513-255-7615
Moravec	Mr. Kipton S.	Texas Instruments	011498161804753
Moreno	Maj Abel	USAF	617-271-4720
Moroney	Mr. Joseph M.	RCA Corp, Electronic Sys Div	609-722-4256
Morris	Mr. Lee G.	VEDA, Inc.	215-672-3200
Morris	Mr. William M.	Naval Research Lab	202-767-5976
Mullen	Mr. Ronald E.	SAIC	703-749-8709
Müllens	Mr. David G.	Boeing Aerospace	206-773-3416
Muller	Mr. James R.	Synetics Corporation	315-337-3510
Mussmann	Mr. David E.	Johns Hopkins Univ/APL	301-953-5000
Mutchler	Dr. Carl N.	The MITRE Corp.	703-883-5527
Muzik	Linda M.	INFOSEC Awareness	301-688-8744
Nakamura	Mr. Yukio	Jet Propulsion Laboratory	818-354-7344
Nellans	Gale D.	Geodynamics Corp	703-971-9000
Nelson	Mr. James B.	Lockheed Missile & Space Co., Inc.	512-448-9488

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Nelson	Mr. Newton P.	Boeing Aerospace Co.	206-655-8544
Ng	Mr. Bin M.	HQ Electronic Security Command	512-925-1601
Nguyen	Mr. Huan H.	Naval Surface Warfare Center	202-394-1456
Nichols	Mr. James R.	Naval Weapons Center	619-939-2335
Noble	Dr. David F.	Engineering Research Associates	703-734-8800
Noble	Dr. Steven S.	Boeing Military Airplane Co.	206-241-3426
Norseen	Mr. John D.	Unisys - Defense Systems	703-620-7860
Northrup, IV	Mr. Francis B.	Intelligence Community Staff	202-376-5568
O'Leary	Mr. Bryan F.	General Dynamics - Electronics	619-573-7718
O'Neill	Mr. Patrick J.	US Army Material Systems Analysis	301-278-6429
Oates	CDR John S.	Naval Intelligence Automation Center	301-763-3514
Ollove	Mrs. Elizabeth	Department of Defense	301-688-7162
Otts	Mr. Ralph H.	UNISYS Corp	612-681-6207
Paasch	Mr. John	Raytheon Co.	617-274-2748
Padgett	Mr. Philip J.	McDonnell Douglas Astronautics Co.	703-276-4618
Parisi	Mr. Michael	JJM Systems, Inc.	215-672-3660
Parsons	Mr. Ronald C.	GTE Government Systems	301-294-8627
Pendergast	Mr. Stephen L.	Hughes Aircraft Co.	714-732-2579
Perkins	Mr. James A.	McDonnell Douglas Corporation	703-247-3513
Perras	CDR Wayne I.	CINCPACFLT	808-471-0759
Perry	Mr. James L.	SENSIS Corporation	315-682-7777
Pfister	Mr. Gerhard	ITT Gilfillan	818-988-2600
Phibbs	Mr. Kenneth W.	Westinghouse D & E Center	301-379-1008
Pieramico	Mr. Alan F.	Technology Service Corp.	203-268-1249
Pinto	Mr. Robert W.	TASC	617-942-2000
Plantenga	Mr. Todd D.	Magnavox Electronic Systems Co	219-429-6707
Powel	Mr. Donald R.	McDonnell Douglas Astronautics Co.	714-896-1493
Pozza, USMC (Ret.)	Colonel John B.	AFCEA	703-631-6238
Prentiss	Dr. Robinson N.	Science Applications Int'l. Corp.	714-640-8662
Presley, Jr.	Dr. Joe A.	ORINCON Corp.	619-455-5530
Presley, Jr.	Dr. Jos. A.	ORINCON Corporation	619-455-5530
Price	Dr. Edward L.	FMC Corp.	703-663-9291
Price	Mr. Charles F.	TASC	617-942-2000
Price	Mr. Charles F.	TASC	617-942-2000
Priebe	Mr. Carey E.	NOSC	619-553-4048
Radcliffe	Mr. Richard H.	Joint Tact. Fusion Prg. Mgt. Office	703-556-2952
Rawicz	Mr. Harris C.	Lockheed Electronics Co., Inc.	201-757-1600
Rawsthorne	Dr. Daniel A.	BDM	703-848-6338
Rebelein	Mr. Paul R.	Honeywell Inc. Systems & Research Ctr.	612-782-7059
Rebovich	Mr. George	The MITRE Corporation	617-271-8261
Reedy	Mr. Robert W.	Hughes Aircraft Co.	213-334-4138
Reilly	Mr. John F.	PAR Government Systems Corp.	315-738-0660
Reuther	Mr. Clifford S.	TRW	703-876-4153
Ritchie	Mr. Curtis A.	E-Systems	214-272-0515
Ritter	Mr. Albert D.	NAVSEASYS COM	202-692-9768
Rivera	Mr. Jorge J.	The Johns Hopkins University	
Robinson	Ms. Joyce M.	Interstate Electronic Co.	714-758-4175
Rock	Mr. Carlton D.	UNISYS, Defense Systems	612-681-6252
Rogers	Mr. Roland D.	Interstate Electronics Corp.	714-758-4072
Rogers	Mr. Steven A.	Lockheed Austin Division	512-448-5236
Rolita, Jr.	Mr. Edward P.	Rolita Consultants	703-791-5505

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Romero	Mr. Joseph	GTE Government Systems	415-966-3156
Romie	Mr. Roger E.	AFWAL/AAWD-1	513-255-6648
Roney	Mr. Philip M.	General Electric Co (MC3I)	215-354-5110
Rosen	Dr. Julie A.	SAIC	703-556-7354
Rothman	Mr. Richard G.	Navy Dept. SPAWAR	202-692-8921
Rothrock	Col John	USAF/INXX	202-695-9066
Ruland	Mr. Timothy P.	Westinghouse Defense Center	301-379-1005
Russo	LCDR Mary S.	Defense Comm. Agency/Advanced Tech Off	703-437-2506
Salisbury	Mr. Robert A.	Tele-Force Associates, Inc.	301-961-6555
Samaan	Dr. Jacob E.	IBM Federal Systems Division	703-841-7300
Sando	2nd LT Kermit A	USAF	513-255-7615
Santa	Mr. Joseph E.	TRW Defense Systems Grou[213-217-3081
Schlegel	Mr. Paul T.	Naval Air Systems Command	202-692-2511
Schneberger	CDR Scott L.	Navy-Chief of Naval Operations	202-694-0297
Schneider	Mr. Raymond J.	Unisys	703-528-3902
Schweiter	Dr. Gail Ann	Daniel H. Wagner, Assoc.	215-644-3400
Scott	Mr. Godfrey L.	FMC Corp.	703-663-9291
Scott	Mr. Godfrey L.	FMC Corporation	703-663-9291
Scott	Mr. Peter A.	Ford Aerospace & Communications Corp.	714-720-6752
Scully	Dr. John W.	U. S. Army Harry Diamond Labs	202-394-2300
Seals	Mr. J. Dennis	AT&T Bell Laboratories	201-386-3373
Selzer	Mr. Fred	Ford Aerospace Corp.	714-720-6283
Shade	Mr. Robert A.	Hazeltine Corp.	516-261-7000
Shankland	Dr. Donn G.	Boeing Computer Services	206-865-3520
Sherman	Mr. Frederick W	General Dynamics/Valley Sys. Div.	714-945-8460
Shirazi	Mr. Mehdi	AF Wright Aeronautical Labs.	513-255-5987
Shoenfeld	Dr. Peter	SAIC	703-821-4466
Silagyi	Mr. Ernest S.	Raytheon Co.	617-271-1040
Silbert	Mr. Mark E.	Naval Air Development Center	215-441-2556
Silverberg	Ms. Marjorie H	General Electric Co.	215-354-3179
Silvey	Mr. Paul E.	The MITRE Corp.	617-271-3497
Simmen	Mr. Robert L.	Radar Control Systems (RADCON)	415-484-4066
Simpson	Mr. John W.	Allied-Signal	301-583-4499
Singer	Mr. Nicholas C.	E-Systems CAPA	703-352-0300
Sipos	Mr. Jon C.	Allied-Signal Inc.	301-583-4112
Smith	Mr. Richard H.	Ford Aerospace Corp.	415-852-4784
Smith	Mr. W. Terry	GTE Government Systems	415-694-1823
Solomon	Mr. Sterling I.	The Analytic Sciences Corporation	703-558-7400
Somoano	Mr. Robert B.	Jet Propulsion Laboratory	818-354-2213
Stamberger	Mr. Frederick J	Hughes Aircraft Company	714-732-1210
Stapleton	Mr. Ronnie A.	Naval Surface Warfare Center	703-663-8906
Staton	Mr. Robin R.	Naval Surface Warfare Ctr.	703-663-7911
Steinberg	Mr. Alan N.	The Analytic Sciences Corp.	704-734-4100
Steinberg	Mr. Richard A.	Johns Hopkins University	
Steinhacker	Mr. Mark	Unisys Corp.	516-574-3317
Steinley	Mr. Michael M.	Naval Air Development Center	215-441-7038
Stephen	Dr. Dick G.	Hazeltine Corp.	516-261-7000
Stepp	Mr. William E.	Lockheed Aeronautical Systems Co.	818-847-5562
Stiglitz	Dr. Irvin G.	MIT Lincoln Laboratory	617-981-7440
Stoltz	Mr. John R.	Advanced Systems Concepts, Inc.	503-386-2225
Storck	Mr. Clarence E.	The MITRE Corp.	619-223-2307

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Studeman	RADM William O.	Director of Naval Intelligence	202-695-3944
Sullivan	Mr. John F.	The MITRE Corp.	617-271-5624
Sutherlin	LCDR Charles T.	Space & Naval Warfare Systems Cmd.	202-692-9113
Suycott	LT Mark L.	Pacific Missile Test Center	805-989-8977
Tasker	Mr. Taymond	Tasker Assoc.	213-472-3869
Teal	Mr. James A.	Harris Corporation	305-984-6448
Terzian	Mr. Richard C.	TRW Defense Systems Group	213-217-4799
Thompson	Mr. James P.	Eaton Corporation/AIL Div.	516-385-2335
Thomsen	CAPT Kurt E.	Rome Air Development Center	315-330-4049
Tolbert	Mr. David C.	Martin Marietta Aero & Naval Systems	301-682-3046
Toman	Mr. Donald J.	Loral Electronic Systems	914-964-3878
Torre	Mr. Frank	Hazeltine Corp.	516-261-7000
Travis	Mr. Tim E.	TRW	213-297-3132
Trenchard	Dr. Herbert A.	Westinghouse Electric Corp.	301-765-7503
Tuttle	Dr. Paul G.	TRW	703-876-8751
Van Doren	Mr. Glennon L.	National Security Agency	301-859-4827
Vanden Dries	Mr. Paul E.	USAF	513-255-9956
Vannicola	Dr. Vincent C.	Rome Air Development Center	315-330-4437
Ventriglio	Mr. Frank J.	Naval Sea Systems Comand	202-692-9538
Vogt	Mr. Gregory S.	Hughes Aircraft Co.	213-334-2598
Vu	Mr. Cuong M.	Naval Surface Warfare Center	202-394-4116
Waagen	Mr. Don E.	Naval Ocean Systems Center	619-553-4046
Wallace	Dr. Timothy P.	M.I.T. Lincoln Laboratory	617-981-2848
Waller	Mr. John S.	HQ TRADOC	804-727-3273
Waltz	Mr. Edward L.	Allied-Signal Inc.	301-583-4245
Wanless	Mr. Loren E.	Lockheed Missiles & Space Co.	512-448-5774
Wenocur	Dr. Roberta D.	Daniel H. Wagner Assoc., Inc.	215-644-3400
Werner	Mr. Keith M.	GE/RCA-Electronic Systems Department	609-866-7704
Wesley	Ms. Debora M.	Westinghouse Electric Corp.	301-765-6142
White	Mr. Franklin E.	Naval Ocean Systems Center	619-553-4036
Whitlow	Mr. Bert L.	ASD/AEIE Navy	513-255-7615
Whittenberger	COL Steven J.	HQ USAF/XOOR	202-697-2475
Widdoss	Mr. Monte G.	Surface Warfare Systems	612-687-1707
Wiener	Dr. Howard L.	Office, Chief of Naval Oper.	202-694-4770
Wiener, II	Dr. Daniel F.	University of Virginia	804-924-7542
William	Mr. Kelly T.	Analytics	703-359-2850
Williams	Mr. Carl E.	Martin Marietta Info. & Comm. Sys.	303-977-0958
Williams	Mr. Elmer F.	Naval Research Lab	202-767-3569
Wilson	Dr. Charles J.	Penn State University	814-863-4102
Windsor	Mr. Stephen W.	TRACOR Aerospace,, Inc.	512-926-2800
Wing	Dr. David H.	Naval Research Lab.	202-767-9829
Winkler	Mr. Gary L.	Analytics	703-359-2850
Wishner	Dr. Richard P.	Advanced Decision Systems	415-960-7300
Wiss	Mr. Victor R.	Naval Surface Warfare Center	703-663-8906
Wofford	Mr. Floyd C.	US Army Ballistic Research lab.	301-278-6657
Wohl	Dr. Joseph G.	ALPHATECH, Inc.	617-273-3388
Wolf	Mr. Richard G.	Westinghouse Defense Center	301-379-1023
Wolff	Mr. Thomas O.	Unisys	612-681-6788
Womack	Mr. Rodney A.	GE/RCA Electronic Systems Division	609-722-7993
Woods	Mr. Edward C.	Westinghouse Electric	301-765-2463
Workman	Mr. Billy J.	The MITRE Corp.	512-675-9640
Wright	Mr. David L.	Naval Sea Systems Command	202-692-3796

LAST NAME	FIRST NAME	COMPANY	PHONE NUMBER
Wu	Mr. Kepi	DoD, Strategic Defense Initiative	202-693-1530
Wuenschel	Mr. Edward F.	GTE Government Systems	415-966-3887
Yanek	Mr. Stephen P.	Johns Hopkins University	301-953-6316
Yannone	Mr. Ronald M.	General Electric (INEWS JVT)	603-885-3736
Youngberg	Mr. Dean A.	HQ LABCOM	301-677-3376
Zabriskie	Mr. George R.	CAS, Inc.	205-895-8600
Zak	Mr. John W.	TRW	
Zentner	Mr. James R.	USAF/F-15 SPO	513-255-4921
Zimmerman	Mr. Jeffrey M.	Technology for Communication Int'l.	301-796-7300